

PCI Express Architecture Configuration Space

Test Specification

Revision 3.0

June 6, 2013



Revision	Revision History	Date
1.0	Initial release.1.0 Release	04/26/2004
1.1	Updated for PCI Express 1.1 test updates/additions.	12/03/2006
2.0	Updated for PCI Express 2.0.	08/25/2008
2.0a	Updated to include 2.0 Test Specification errata and changes in the Errata for the PCI Express Base Specification Revision 2.0 document. Also typographical, grammatical, and formatting errors were corrected.	06/19/2009
3.0	Updated for PCI Express 3.0. Added tests for ATS, and SR-IOV specifications. Obsoleted some motherboard-specific tests.	03/20/2013

PCI-SIG® disclaims all warranties and liability for the use of this document and the information contained herein and assumes no responsibility for any errors that may appear in this document, nor does PCI-SIG make a commitment to update the information contained herein.

Contact the PCI-SIG office to obtain the latest revision of this specification.

Questions regarding this specification or membership in PCI-SIG may be forwarded to:

Membership Services

www.pcisig.com

E-mail: administration@pcisig.com

Phone: 503-619-0569

Fax: 503-644-6708

Technical Support

techsupp@pcisig.com

DISCLAIMER

This PCI Code and ID Assignment Specification is provided *as is* with no warranties whatsoever, including any warranty of merchantability, non-infringement, fitness for any particular purpose, or any warranty otherwise arising out of any proposal, specification, or sample. PCI-SIG disclaims all liability for infringement of proprietary rights, relating to use of information in this specification. No license, express or implied, by estoppel or otherwise, to any intellectual property rights is granted herein.

PCI Express is a trademark of PCI-SIG.

All other product names are trademarks, registered trademarks, or service marks of their respective owners.

Contents

1. INTRODUCTION.....	6
1.1. DOCUMENT CONVENTIONS	6
1.2. COVERAGE	7
1.3. TEST OPTIONS FOR SPECIFICATION REVISIONS.....	8
1.4. FUNCTION UNDER TEST DISCOVERY	9
1.4.1. <i>Discovery of Basic Functions, ARI Functions, Base Functions,</i> <i>Physical Functions, and Virtual Functions</i>	<i>9</i>
2. TEST DESCRIPTIONS	13
2.1. COMMON PROCEDURES AND ROUTINES.....	13
2.1.1. <i>Standard Initialization Procedure.....</i>	<i>13</i>
2.1.2. <i>Standard Register Field Characteristic Test Routines.....</i>	<i>21</i>
2.2. CONFIGURATION REGISTER TESTS (ALL DEVICES).....	43
2.2.1. <i>TD_1_2 PCI Express Capability Structure.....</i>	<i>43</i>
2.2.2. <i>TD_1_3 PCI Express Capabilities Register.....</i>	<i>45</i>
2.2.3. <i>TD_1_4 Device Capabilities, Device Control, and Device Status Registers.....</i>	<i>47</i>
2.2.4. <i>TD_1_40 Device Capabilities 2, Device Control 2, and</i> <i>Device Status 2 Registers (PCIe Cap Ver = 2).....</i>	<i>51</i>
2.2.5. <i>TD_1_5 Link Capabilities, Link Control, and Link Status Registers</i>	<i>57</i>
2.2.6. <i>TD_1_41 Link Capabilities 2, Link Control 2, and Link Status 2 Registers</i> <i>(PCIe Cap Ver = 2).....</i>	<i>62</i>
2.2.7. <i>TD_1_49 Slot Capabilities, Slot Control, and Slot Status Registers</i> <i>(PCIe Cap Ver = 2).....</i>	<i>68</i>
2.2.8. <i>TD_1_50 Slot Capabilities 2, Slot Control 2, and Slot Status 2 Registers</i> <i>(PCIe Cap Ver = 2).....</i>	<i>73</i>
2.2.9. <i>TD_1_51 Root Capabilities, Root Control, and Root Status Registers</i> <i>(PCIe Cap Ver = 2).....</i>	<i>74</i>
2.2.10. <i>TD_1_6 MSI Capability Structure.....</i>	<i>77</i>
2.2.11. <i>TD_1_7 Advanced Error Reporting Extended Capability Structure</i>	<i>79</i>
2.2.12. <i>TD_1_8 Virtual Channel Extended Capability Structure</i>	<i>99</i>
2.2.13. <i>TD_1_9 Device Serial Number Extended Capability Structure</i>	<i>104</i>
2.2.14. <i>TD_1_10 Power Budgeting Extended Capability Structure</i>	<i>105</i>
2.2.15. <i>TD_1_11 Command and Status Registers</i>	<i>109</i>
2.2.16. <i>TD_1_12 Cache Line Size, Master/Primary Latency Timer, and</i> <i>Min_Gnt/Max_Lat Registers</i>	<i>111</i>
2.2.17. <i>TD_1_13 Interrupt Pin and Interrupt Line Registers.....</i>	<i>113</i>
2.2.18. <i>TD_1_14 Secondary Latency Timer and Secondary Status Registers.....</i>	<i>114</i>
2.2.19. <i>TD_1_15 Bridge Control Register</i>	<i>116</i>
2.2.20. <i>TD_1_16 PCI Power Management Capability Structure.....</i>	<i>118</i>
2.2.21. <i>TD_1_17 MSI-X Capability Structure</i>	<i>124</i>
2.2.22. <i>TD_1_18 Base Address Registers.....</i>	<i>127</i>
2.2.23. <i>TD_1_19 Multi-Function Virtual Channel Extended Capability Structure</i>	<i>130</i>
2.2.24. <i>TD_1_52 Vendor Specific Capability Structure</i>	<i>135</i>
2.2.25. <i>TD_1_20 Vendor-Specific Extended Capability Structure.....</i>	<i>136</i>

2.2.26.	<i>TD_1_21 BIST Register</i>	<i>138</i>
2.2.27.	<i>TD_1_22 Slot Numbering Capability Structure.....</i>	<i>140</i>
2.2.28.	<i>TD_1_53 SSID/SSVID Capability Structure.....</i>	<i>141</i>
2.2.29.	<i>TD_1_23 PCI Next Capability Pointer Register</i>	<i>143</i>
2.2.30.	<i>TD_1_24 PCI Express Next Extended Capability Pointer Register</i>	<i>144</i>
2.2.31.	<i>TD_1_25 Misc Type 0 Config Space Header Registers.....</i>	<i>145</i>
2.2.32.	<i>TD_1_26 Misc Type 1 Config Space Header Registers.....</i>	<i>148</i>
2.2.33.	<i>TD_1_27 Multi-Function.....</i>	<i>155</i>
2.2.34.	<i>TD_1_28 Vital Product Data Capability Structure</i>	<i>162</i>
2.2.35.	<i>TD_1_32 PCI-X Capability Structure</i>	<i>165</i>
2.2.36.	<i>TD_1_30 Root Complex Link Declaration Extended Capability Structure</i>	<i>168</i>
2.2.37.	<i>TD_1_31 Root Complex Internal Link Control Extended Capability Structure.....</i>	<i>172</i>
2.2.38.	<i>TD_1_34 RCRB Header Extended Capability Structure.....</i>	<i>175</i>
2.2.39.	<i>TD_1_29 Root Complex Event Collector Endpoint Association Extended Capability Structure</i>	<i>177</i>
2.2.40.	<i>TD_1_35 Configuration Access Correlation Extended Capability Structure</i>	<i>179</i>
2.2.41.	<i>TD_1_39 Function Level Reset.....</i>	<i>180</i>
2.2.42.	<i>TD_1_42 ACS Extended Capability Structure.....</i>	<i>183</i>
2.2.43.	<i>TD_1_43 ARI Extended Capability Structure</i>	<i>186</i>
2.2.44.	<i>TD_1_44 DPA Extended Capability Structure</i>	<i>189</i>
2.2.45.	<i>TD_1_45 Resizable BAR Extended Capability Structure</i>	<i>192</i>
2.2.46.	<i>TD_1_46 Multicast Extended Capability Structure.....</i>	<i>196</i>
2.2.47.	<i>TD_1_47 LTR Extended Capability Structure</i>	<i>199</i>
2.2.48.	<i>TD_1_48 TPH Requester Extended Capability Structure</i>	<i>201</i>
2.2.49.	<i>TD_1_54 ATS Extended Capability Structure.....</i>	<i>204</i>
2.2.50.	<i>TD_1_55 Page Request Extended Capability Structure.....</i>	<i>206</i>
2.2.51.	<i>TD_1_56 SR-IOV Extended Capability Structure.....</i>	<i>208</i>
2.2.52.	<i>TD_1_57 MR-IOV Extended Capability Structure</i>	<i>214</i>
2.2.53.	<i>TD_1_58 Secondary PCI Express Extended Capability Structure</i>	<i>216</i>
2.2.54.	<i>TD_1_59 Protocol Multiplexing Extended Capability Structure</i>	<i>219</i>
2.3.	CONFIGURATION REGISTER TESTS (DEVICES WITH UPSTREAM PORTS).....	225
2.3.1.	<i>TD_2_1 ASPM Configuration Stress (Upstream Ports).....</i>	<i>225</i>
2.3.2.	<i>TD_2_2 Link Training Stress</i>	<i>226</i>
2.3.3.	<i>TD_2_3 Tolerance of Hot-Plug Signaling/Ignored Messages.....</i>	<i>229</i>
2.3.4.	<i>TD_2_4 Response To Earliest Allowed Configuration Requests After Reset</i>	<i>230</i>
2.3.5.	<i>TD_2_5 Response to Different Bus and Device Numbers.....</i>	<i>232</i>
2.3.6.	<i>TD_2_6 Secondary Bus Reset (Upstream Ports).....</i>	<i>233</i>
2.3.7.	<i>TD_2_7 Requested Link Speed (Upstream Ports).....</i>	<i>234</i>
2.3.8.	<i>TD_2_8 Supported Link Width (Upstream Ports)</i>	<i>238</i>
2.3.9.	<i>TD_2_9 Software Requested Link Equalization (Upstream Ports)</i>	<i>240</i>
2.4.	CONFIGURATION REGISTER TESTS (DEVICES WITH DOWNSTREAM PORTS)	244
2.4.1.	<i>TD_3_1 Slot Capabilities, Slot Control, and Slot Status Registers (PCIe Cap Ver = 1).....</i>	<i>244</i>
2.4.2.	<i>TD_3_2 Root Capabilities, Root Control, and Root Status Registers (PCIe Cap Ver = 1).....</i>	<i>249</i>
2.4.3.	<i>TD_3_15 ASPM Configuration Stress (Downstream Ports)</i>	<i>251</i>
2.4.4.	<i>TD_3_13 Secondary Bus Reset (Downstream Ports).....</i>	<i>253</i>
2.4.5.	<i>TD_3_10 Initiated Link Speed (Downstream Ports).....</i>	<i>254</i>

2.4.6.	<i>TD_3_11 Supported Link Width (Downstream Ports)</i>	258
2.4.7.	<i>TD_3_12 ARI Downstream Ports Support non-Zero Device Number</i>	261
2.4.8.	<i>TD_3_14 Software Initiated Link Equalization (Downstream Ports)</i>	262
APPENDIX A. INFORMATIONAL TEST DETAILS		266
A.1	TD_1_57X MR-IOV EXTENDED CAPABILITY STRUCTURE	266
APPENDIX B. OBSOLETE TEST DETAILS		347
B.1	TD_1_1.....	347
B.2	TD_1_33 ROOT COMPLEX INTEGRATED ENDPOINT.....	347
B.3	TD_1_35X CONFIGURATION ACCESS CORRELATION EXTENDED CAPABILITY STRUCTURE.....	347
B.4	TD_1_36 TRUSTED CONFIG SPACE HEADER.....	349
B.5	TD_1_37 CONFIGURATION ACCESS CORRELATION TRUSTED CAPABILITY STRUCTURE	350
B.6	TD_1_38 VENDOR-SPECIFIC TRUSTED CAPABILITY STRUCTURE	352
B.7	TD_3_3 ACCURATE SLOT REPORTING	354
B.8	TD_3_4 BASIC HOT-PLUG INSERTION.....	355
B.9	TD_3_5 BASIC HOT-PLUG REMOVAL.....	361
B.10	TD_3_6 BASIC HOT-PLUG SURPRISE REMOVAL	365
B.11	TD_3_7 ATTENTION BUTTON MRL INDICATOR CONTROL	367
B.12	TD_3_8 LINK RETRAINING STRESS.....	374
B.13	TD_3_9 SLOT CAPABILITIES 2, SLOT CONTROL 2, AND SLOT STATUS 2 REGISTERS	376
APPENDIX C. ACKNOWLEDGEMENTS.....		377

1. Introduction

This document primarily covers PCI Express testing of all defined PCI Express Device Types and RCRBs for the standard Configuration Space mechanisms, registers, and features in Chapter 6 of the PCI Local Bus Specification and Chapter 7 of the *PCI Express Base Specification* (some additional tested registers are described in other specifications that are referenced in the individual test description). This specification does not describe the full set of PCI Express tests for these devices. In particular, devices must also meet the requirements and tests described in the latest versions of the following documents as well as any other tests provided by the PCI-SIG:

PCI Express Architecture PHY Test Specification

PCI Express Architecture Platform Init/Config Test Specification

PCI Express Architecture Link Layer and Transaction Layer Test Specification

The test descriptions can be referenced to obtain specific details on how the specific registers will be tested. Test descriptions also list which device types are tested. Test descriptions include “The test fails if” sub-sections that provide a list of some of the specific conditions that would cause the test to fail. Test descriptions include a reference to the applicable specifications and also provide the specification revision.

1.1. Document Conventions

Capitalization

Register names and the names of fields and bits in registers and headers are presented with the first letter capitalized and the remainder in lower case. Multiple word names have the first letter of each word capitalized.

Numbers and Number Bases

Hexadecimal numbers are written with a lower case “h” suffix, (e.g., FFFh and 80h). Hexadecimal numbers larger than four digits are represented with a space dividing each group of four digits, as in 1E FFFF FFFFh.

Binary numbers are written with a lower case “b” suffix, (e.g., 1001b and 10b). Binary numbers larger than four digits are written with a space dividing each group of four digits, as in 1000 0101 0010b.

Numbers with an “x” in them indicate that the position of the “x” in the number is a don’t care value and can have any legal value for that number base (e.g., 1x00b represents both 1000b and 1100b).

All numbers without a suffix are decimal.

1.2. Coverage

The Configuration Space Test Consideration for the PCI Express Architecture covers items in Chapter 6 of the PCI Local Bus Specification and Chapter 7 of the *PCI Express Specification* and configuration items from other specifications that apply to PCI Express devices.

Revision 1.0 of this document includes updates based on the *PCI Express Base Specification Revision 1.0a*.

Revision 1.1 of this document includes updates based on the *PCI Express Base Specification Revision 1.1*.

Revision 2.0 of this document includes updates based on the *PCI Express Base Specification Revision 2.0*.

Revision 2.0a of this document includes updates based on the document *Errata for the PCI Express Base Specification Revision 2.0*, dated Feb 27, 2009. For 2.0 compliance testing, two interpretations will be accepted, the original Base 2.0 specification requirement, and the updated Base 2.0 requirement that includes the 2.0 errata document. For future compliance testing, only the updated interpretation will be accepted.

Revision 3.0 of this document includes updates based on the *PCI Express Base Specification Revision 2.1*. (This is equivalent to *PCI Express Base Specification Revision 2.0* (including Base 2.0 errata) plus all applicable ECNs to 2.0). In addition to this, it also includes updates based on various ECNs to the *PCI Express Base Specification Revision 2.1*. Since the ECNs to Base 2.1 are optional, a 2.1 compliant device may choose to implement or not implement any or all of these ECNs to Base 2.1. For 2.1 compliance testing, a device that implements any of these released ECNs to Base 2.1 will be tolerated by the test and will be considered to pass if it implements those specific ECNs to Base 2.1 correctly. This revision of this document also includes the *Single Root I/O Virtualization and Sharing Specification Revision 1.0* and any Base 1.x or Base 2.x device that implements SR-IOV features must comply with this revision of the specification.

Revision 3.0 of this document includes updates based on the *PCI Express Base Specification Revision 3.0*. This revision of this document also includes updates to the *Single Root I/O Virtualization and Sharing Specification Revision 1.1*. A Base 3.x device that implements SR-IOV features must comply with this revision of the specification. A Base 2.x device that implements SR-IOV features may comply with either this revision of the specification or the previous *Single Root I/O Virtualization and Sharing Specification Revision 1.0*. A Base 1.x device that implements SR-IOV features must comply with the previous *Single Root I/O Virtualization and Sharing Specification Revision 1.0*.

Note: Register names and fields reflect those listed in the Base Specification revision applicable to the test document revision. The only exception is the Link Capabilities register bits 3-0, which are listed in this document as “Max Link Speed/Supported Link Speeds”, to reflect its changed definition from Base 2.x to Base 3.x.

1.3. Test Options for Specification Revisions

The Configuration Space Test Consideration for the PCI Express Architecture covers several different revisions of the Base specification. A user selectable menu option is available to configure the test to a specific Base specification revision. The requirements for each of these individual menu options may change with each revision of this document, as may the number of menu options. The menu options and their requirements are stated below.

If implemented by the DUT, ECNs to the indicated Base specification will be tested and will result in a fail if the DUT does not comply with the ECN. If not implemented by the DUT, ECNs to the indicated Base specification will not affect pass/fail. The ECNs are listed in the individual test descriptions.

“Test against either 1.1 or 2.0 Spec”

This selection will consider a DUT to pass if it complies with the following (including either of the following Base Specifications):

- ☐ *PCI Express Base Specification Revision 1.1 with Errata for the PCI Express Base Specification Revision 1.1* (dated Feb 8, 2007)
- ☐ *PCI Express Base Specification Revision 2.0 with Errata for the PCI Express Base Specification Revision 2.0* (dated Feb 27, 2009)
- ☐ *Single Root I/O Virtualization and Sharing Specification Revision 1.0* (optional)

“Test against 2.0 Spec Only”

This selection will consider a DUT to pass if it complies with the following:

- ☐ *PCI Express Base Specification Revision 2.0 with Errata for the PCI Express Base Specification Revision 2.0* (dated Feb 27, 2009)
- ☐ *Single Root I/O Virtualization and Sharing Specification Revision 1.0* (optional)

“Test against 2.1 or 2.0 Spec”

This selection will consider a DUT to pass if it complies with the following (including either of the following Base Specifications):

- ☐ *PCI Express Base Specification Revision 2.0 with Errata for the PCI Express Base Specification Revision 2.0* (dated Feb 27, 2009)
- ☐ *PCI Express Base Specification Revision 2.1*
- ☐ *Single Root I/O Virtualization and Sharing Specification Revision 1.1 or Revision 1.0* (optional)

“Test against 3.0 Spec Only”

This selection will consider a DUT to pass if it complies with the following:

- ☐ *PCI Express Base Specification Revision 3.0*
- ☐ *Single Root I/O Virtualization and Sharing Specification Revision 1.1* (optional)

1.4. Function Under Test Discovery

The test program shall test hardware at the function level of granularity. This means that the test will run on an individual function identified by a unique Bus Number/Device Number/Function Number address. The test program shall provide the user with a selection prompt to allow them to choose the specific function to run the test on (henceforth referred to as the function under test). The user selection prompt will group each function under test by its reported Device/Port Type (PCI Express Capability).

1.4.1. Discovery of Basic Functions, ARI Functions, Base Functions, Physical Functions, and Virtual Functions

The test program shall scan all supported Bus Numbers (0-256), all supported Device Numbers (0-31), and all supported Function Numbers (0-7).

(Note: Some systems limit the maximum number of supported buses to less than 256. The test program shall use appropriate SBIOS ACPI tables to determine the maximum number of supported buses before beginning the scan.)

(Note: All Function Numbers shall be scanned regardless of the setting of the Multi-function bit in the Header Type. This ensures that functions that do not correctly implement the Multi-function bit will still be selectable as a function under test.)

A device that supports ARI may support up to 256 functions. In order to do this, the standard Device Number field is no longer used, and the Function Number field now goes from 0-255. (Test software can still use the Device Number field as a concept to allow the same algorithm to handle both non-ARI and ARI functions.)

The test program creates a 16 bit Bus Number/Device Number/Function Number value [BDF] and increments it by 1 for each pass through the scan. The Bus Number/Device Number/Function Number value [BDF] consists of: bits 15-8 = Bus Number; bits 7-3 = Device Number; bits 2-0 = Function Number.

A physical function (PF) that support SR-IOV may contain virtual functions (VFs) that are hidden by default. Once the PF is programmed to enable the VFs, there may be up to 65536 VFs visible. VFs require special VF aware detection algorithms, as they always return Vendor ID of FFFFh and Device ID of FFFFh. Any number of VFs greater than 256, require additional Bus Numbers to be available to the VF, and the bus hierarchy must be programmed to route the additional bus numbers to the path leading to the VFs.

The scan starts with [BDF] as 0000h.

- a. Read back a WORD at location 00h (Vendor ID register) in Configuration Space, and if 0001h (CRS Software Visibility notification) is found, continue reading this location until a value other than 0001h is found. If a value other than 0001h is not found within 1 second, then this [BDF] location is broken (though it could be detected later as part of PF detection) and the scan restarts at step a) with the next [BDF] value.
- b. Read back a WORD at location 00h (Vendor ID register) in Configuration Space, and if FFFFh is found, this [BDF] location is empty (though it could be a VF which will be detected later as part of PF detection) and the scan restarts at step a) with the next [BDF] value.

- c. Examine the Capability ID field for each of the function's Capabilities. Determine how many instances of the Capability ID of 10h (PCI Express Capability) are found. If none are found, this [BDF] location is not a PCI Express function and the scan restarts at step a) with the next [BDF] value.
- d. Having found the PCI Express Capability at this [BDF], read a DWORD at offset 02h (PCI Express Capabilities register) in the PCI Express Capability and use the Device/Port Type field value to identify the PCI Express type that the function will be listed under in the user selection prompt. (Note: Any functions reporting reserved encodings for Device/Port Type must be listed under the category "Unknown" in the user selection prompt. These can still be selected by the user as the function under test.)
- e. If the current Function Number is 0, then examine the Extended Capability ID field for each of the detected Extended Capabilities. Determine how many instances of the Extended Capability ID of 000Eh (ARI Extended Capability) are found. If none are found, this [BDF] location does not contain an ARI Extended Capability, so skip to step g).
- f. If an ARI Extended Capability was found, then find the immediate downstream port connected to this function (the device at the other end of the link for this function) and write ARI Forwarding Enable (Device Control 2 register) to 1. (This will allow an ARI capable downstream port to forward all Device Numbers on this port. This will cause any ARI functions on this link to become detectable by test software. Setting this bit on a non-ARI capable downstream port will not cause any change since this bit would not be implemented on such a port.) Note: It is important that this setting remain unchanged for this port for the remainder of all testing, otherwise the ARI functions will disappear. If a test specifically causes this bit to be cleared (when it was previously set in this step), the test must restore the set value before proceeding. This is important for any test that would cause any downstream port with this bit set, to revert to its default values.
- g. Examine the Extended Capability ID field for each of the detected Extended Capabilities. Determine how many instances of the Extended Capability ID of 0010h (SR-IOV Extended Capability) are found. If none are found, this [BDF] location does not contain a SR-IOV Extended Capability, so skip to step aa).
- h. If a SR-IOV Extended Capability was found, read a WORD at offset 0Eh (TotalVFs register) in the SR-IOV Extended Capability. If this value is 0000h, this PF does not contain any VFs, so skip to step aa).
- i. If the value read from TotalVFs is non-zero, write this value to the WORD at offset 10h (NumVFs). Then write VF Enable (SR-IOV Control register) to 1. (This will cause all VF functions in this PF between 1 and the value in InitialVFs to become detectable by test software as long as they remain in the Active.Available state. Note: It is important that these settings remain unchanged for this PF for the remainder of all testing, otherwise the VF functions will disappear. If a test specifically causes this field to be changed or this bit to be cleared (when they were previously programmed in this step), the test must restore the programmed values before proceeding. This is important for any test that would cause any PF with this programmed field and this bit set, to revert to its default values.
- j. If the value read from InitialVFs is not less than the value read from TotalVFs, then all VFs are already in the Active.Available state, so skip to step r.

- k. If TotalVFs register returns a value greater than 0 and VF Migration Capable (SR-IOV Capabilities register) returns 1, then read VF Migration State Offset and VF Migration State BIR (both in VF Migration State Array Offset register) and calculate the VF Migration State Array starting address value [MSAD] as follows: $[MSAD] = \text{VF Migration State Array Offset value} + \text{the value returned in the BAR indicated by the value in VF Migration State BIR}$.
- l. After enabling Memory Space access for the PF, set the inactive VF number value [IVFN] to the value in InitialVFs + 1.
- m. If [IVFN] is less than or equal to the value in TotalVFs, read a byte from the Memory Space address given by: $[MSAD] + ([IVFN] - 1)$. Use this value masking out the lower two bits and combine it with the Migration State value of 01b (Dormant.MigrateIn). Write this new byte (bits 7-2 containing read back value, bits 1-0 containing 01b) back to the same address.
- n. Wait 100 ms.
- o. If [IVFN] is less than or equal to the value in TotalVFs, read a byte from the Memory Space address given by: $[MSAD] + ([IVFN] - 1)$. Use this value masking out the lower two bits and combine it with the Migration State value of 11b (Active.Available). Write this new byte (bits 7-2 containing read back value, bits 1-0 containing 11b) back to the same address. (This will cause all VF functions in this PF between InitialVFs and the value in TotalVFs to become detectable by test software as long as they remain in the Active.Available state. Note: It is important that these settings remain unchanged for this PF for the remainder of all testing, otherwise the VF functions will disappear. If a test specifically causes this field to be changed (when they were previously programmed in this step and the previous two steps), the test must restore the programmed values, using the same programming sequence before proceeding. This is important for any test that would cause any PF with this programmed field, to revert to its default values.
- p. Wait 100 ms.
- q. Increment [IVFN] by adding 1, and if the new [IVFN] is less than or equal to the value in TotalVFs, repeat steps m-q).
- r. If the value written to NumVFs is greater than 255, then the test program must ensure that additional bus numbers are available on the immediate downstream port connected to this function (the device at the other end of the link for this function). The Subordinate Bus Number register for the downstream port device must return a value that includes the required number of bus numbers for all the VFs (required number of bus numbers = TotalVFs divided by 255, rounded up to the next integer value). If the number of allocated bus numbers (number of allocated bus numbers = Subordinate Bus Number minus Secondary Bus Number) is not sufficient, additional bus numbers must be allocated to this specific hierarchy (including any other devices in the upstream hierarchy), so that all the VFs may be visible. In doing so, some bus numbers may need to be re-allocated, which may require a partial or complete rescan of the Bus Numbers. Note: It is important that these bus number allocations remain unchanged for the remainder of all testing, otherwise some VF functions will disappear. If a test specifically causes these bus number allocations to change, the test must restore the programmed allocation before proceeding.
- s. If a SR-IOV Extended Capability was found, read a WORD at offset 14h (First VF Offset register) in the SR-IOV Extended Capability.

- t. If a SR-IOV Extended Capability was found, read a WORD at offset 16h (VF Stride register) in the SR-IOV Extended Capability.
- u. Set value [VFN] to 1.
- v. Using the current [BDF] value, calculate the VF location as follows: $[VF\ BDF] = ([BDF] + \text{First VF Offset} + (([VFN]-1) * \text{VF Stride})) \bmod 2^{**} 16$.
- w. Read back a WORD at location 00h (Vendor ID register) in Configuration Space, and if 0001h (CRS Software Visibility notification) is found, continue reading this location until a value other than 0001h is found. If a value other than 0001h is not found within 1 second, then this [VF BDF] location is broken so ignore it and skip to step z.
- x. Examine the Capability ID field for each of the function's Capabilities. Determine how many instances of the Capability ID of 10h (PCI Express Capability) are found. If none are found, this [VF BDF] location is not a PCI Express function so skip to step z).
- y. Having found the PCI Express Capability at this [VF BDF], read a DWORD at offset 02h (PCI Express Capabilities register) in the PCI Express Capability and use the Device/Port Type field value to identify the PCI Express type that the VF will be listed under in the user selection prompt. (Note: Any VFs reporting reserved encodings for Device/Port Type must be listed under the category "Unknown" in the user selection prompt. These can still be selected by the user as the function under test. Any VFs reporting non-Endpoint encodings for Device/Port Type must be listed under their reported category in the user selection prompt. These can still be selected by the user as the function under test.)
- z. Increment [VFN] by 1, and if the new [VFN] is less than the value in NumVFs, repeat steps v-z), still using the current [BDF].
- aa. Increment [BDF] by 1, and if the new [BDF] is less than or equal to the highest Bus Number/Device Number/Function Number supported on this system, repeat steps a-aa).

The user selection prompt shall appear as a list of all the defined Device/Port Type values. Only if a function with a reserved encoding of Device/Port Type is found, an additional "Unknown" type value will also appear.

When the user selects one of these Device/Port Types, a new selection prompt will appear that lists each discovered function for that type, by giving its Bus Number/Device Number/Function Number address. (For a standard display format, the Device Number field can still be displayed for ARI functions, since any downstream device function that is not on Device Number 0, will be an ARI function.)

Functions, BFs, PFs, and VFs can all be listed using the same standard display format, as the user does not need to be able to distinguish between them, since the test will do so as needed by individual test cases.

When the user selects the specific function, that function will be designated as the function under test.

2. Test Descriptions

This document describes only tests that can be run without special purpose test hardware.

2.1. Common Procedures and Routines

The Configuration Space Registers common tests cover mainly the registers defined in Chapter 6 of the *PCI Local Bus Specification* and Chapter 7 of the PCI Express specification. There is more than one state in which many of these tests can be run. Many of the tests are run with the function under test in more than one of these states. Each of the states and procedure used to put the function under test into this state are described here. This information is provided to help with debugging in cases where the function under test is not even reaching the desired starting state for the test. The individual tests mention which states they are run on, but do not repeat the setup procedure description.

2.1.1. Standard Initialization Procedure

Note: Test software blocks the operating system PCI/PCI Express stack from interacting with the function under test once the initialization sequence has started. This means that all the functions in the device containing the function under test, and all functions connected to the link hierarchy either started by the function under test, or for a downstream component, started by the upstream component connected to the function under test, must not have any active software driver. This can be accomplished, by either not installing the software driver, or by using the operating system's device-specific controls to disable the function (e.g., for a Windows operating system, using Device Manager to disable those functions). Additional steps must be taken to ensure that a primary boot device is not used as the function under test (e.g., the primary display device must not be used as the function under test, as it will cause the test's display output to disappear and the operating system to hang).

2.1.1.1 Reset Sequences

The test requires that the function under test, be put into its default state. This is achieved by executing an appropriate Reset Sequence that will cause the function under test to execute its internal hardware initialization, and put the hardware into a defined initial state. Different Reset Sequences are possible under PCI Express, each with small variations in the resulting hardware behaviour.

Each Reset Sequence is defined below, with any special affect on the hardware indicated.

Note: For 2.x and earlier testing, Link Enable/Disable (DL_Down) is the only Reset Sequence used by test software.

For 3.x testing, Link Enable/Disable (DL_Down), Hot Reset (Secondary Bus Reset), and Function Level Reset (FLR) are the only Reset Sequences used by test software.

A user selectable menu option is available to configure the test to a specific Reset Sequence. An additional selection (“Use device type dependent default reset mechanism”) is provide to automatically select the default Reset Sequence based on the DUT’s Device/Port Type. The menu options, their requirements, and default selections are stated below.

2.1.1.1.1 Link Enable/Disable (DL_Down) Reset Sequence

This is the default Reset Sequence for all functions under test that are connected to an upstream port (e.g., Legacy Endpoints, PCI Express Endpoints, Switch Upstream Ports, PCI Express to PCI/PCI-X Bridges) and all functions under test that are controlled by an upstream port component (e.g., Switch Downstream Ports).

1. If the function under test has a Type 1 Configuration Space header, then the following registers are read and their values recorded for later restoration: Primary Bus Number; Secondary Bus Number; Subordinate Bus Number. The values must be recorded for all of the following: a) if the function under test is not a Root Port then the function itself; b) if the function under test is a Switch Downstream Port then the Switch Upstream Port that the Function under test is subordinate to; c) all Switch Ports and Bridges connected below the link hierarchy originating from the function under test; d) all functions behind the Function under test that also have a Type 1 Configuration Space header. (Note: Resetting a Switch Downstream Port or PCI/PCI-X to PCI Express Bridge will cause it to transmit Hot Reset while resetting a PCI Express to PCI/PCI-X Bridge will cause it to assert bus reset, either of which will clear the bus number registers of all Switches and Bridges below it. This will make the Switch and Bridge functions behind the Switch or Bridge disappear, so after the reset completes all these bus numbers must be restored before testing can continue.)
2. If the function under test does not connect to a link, then this is not a failure and the remaining steps are skipped. (Note: For this Device Type, the test will report skipped for any sticky or default value portions of the test, when this type of Reset Sequence is used.)
3. If the function under test is a Switch Downstream Port, the Link Disable field in the Link Control register on the next higher downstream port (the port connected to the link of the Switch Upstream Port that the function under test is immediately subordinate to) is set to 1 using a WORD access, while preserving all the other fields in this register.
4. If the function under test has a non-Switch downstream port, the Link Disable field in the Link Control register on the function under test is set to 1 using a WORD access, while preserving all the other fields in this register. (Note: For this Device Type, the test will report skipped for any sticky or default value portions of the test, when this type of Reset Sequence is used.)
5. If the function under test has an upstream port, the Link Disable field in the Link Control register on the immediate downstream port (the port connected to the link of the function under test) is set to 1 using a WORD access, while preserving all the other fields in this register.
6. After 50 ms the Link Disable field that was previously set in any one of steps 3-5 is cleared to 0 using a WORD access, while preserving all the other fields in this register.
7. Software starts a timer which will be called the Reset Timeout Timer [RTT]. The [RTT] is started with a value of 0. Note: The [RTT] must be capable of counting up to at least 1 second.

8. (Note: In the interest of saving time, this step may be skipped if the function under test does not support 8.0 GT/s.) If the function under test has an upstream port, the immediate downstream port connected to the function under test is checked to see if it has completed training. If the function under test is a Switch Downstream Port, the immediate downstream port connected to the link of the Switch Upstream Port (that the function under test is immediately subordinate to) is checked to see if it has completed training. If the function under test has a non-Switch downstream port, the function under test is checked to see if it has completed training. Firstly the Data Link Layer Link Active Reporting Capable field in the Link Capabilities register is read and if it returns 0, then go on to the next step. If the Data Link Layer Link Active Reporting Capable field returned 1, then continuously read the Data Link Layer Link Active field in the Link Status register, and when it returns 1, go on to the next step. If the Data Link Layer Link Active field does not return 1 before the [RTT] reaches 1 second, then report this as a link training failure, and skip the remaining steps.
9. Test software waits for 100 ms.
10. If the function under test is a Switch Downstream Port, then for the Switch Upstream Port (that the function under test is immediately subordinate to) test software restores all the values recorded in step 1 for: Primary Bus Number; Secondary Bus Number; Subordinate Bus Number.
11. (This step only applies to a function that is not an RCRB.) Read from the function under test, a WORD at location 00h (Vendor ID register) in Configuration Space and if it returns 0001h (CRS Software Visibility notification), then the read is repeated, until a different value is returned. If the different value is not returned before the [RTT] reaches 1 second, then report a function not detected failure, and skip the remaining steps.
12. (This step only applies to a function that is not a VF and not an RCRB.) Read from the function under test, a WORD at location 00h (Vendor ID register) in Configuration Space and if it returns FFFFh, then the read is repeated, until a different value is returned. If the different value is not returned before the [RTT] reaches 1 second, then report a function not detected failure, and skip the remaining steps.
13. If the function under test is a Switch Downstream Port, then steps 8-9 are repeated, by checking the downstream port of the function under test.
14. Stop the [RTT] timer.
15. Test software restores all the values recorded in step 1 for: Primary Bus Number; Secondary Bus Number; Subordinate Bus Number. These individual values must be restored in the Switch, all subordinate Switches and Bridges, and in the Type 1 Configuration Space and any subordinate Type 1 Configuration Spaces.

2.1.1.1.2 Hot Reset (Secondary Bus Reset) Reset Sequence

This is the default Reset Sequence selected automatically, for all functions under test that are connected to a downstream port that are not controlled by an upstream port component (e.g., Root Ports, PCI/PCI-X to PCI Express Bridges).

1. If the function under test has a Type 1 Configuration Space header, then the following registers are read and their values recorded for later restoration: Primary Bus Number; Secondary Bus Number; Subordinate Bus Number. The values must be recorded for all of the following: a) if the function under test is not a Root Port then the function itself; b) if the function under test is

a Switch Downstream Port then the Switch Upstream Port that the Function under test is subordinate to; c) all Switch Ports and Bridges connected below the link hierarchy originating from the function under test; d) all functions behind the Function under test that also have a Type 1 Configuration Space header. (Note: Resetting a Switch Downstream Port or PCI/PCI-X to PCI Express Bridge will cause it to transmit Hot Reset while resetting a PCI Express to PCI/PCI-X Bridge will cause it to assert bus reset, either of which will clear the bus number registers of all Switches and Bridges below it. This will make the Switch and Bridge functions behind the Switch or Bridge disappear, so after the reset completes all these bus numbers must be restored before testing can continue.)

2. If the function under test does not connect to a link, then this is not a failure and the remaining steps are skipped. (Note: For this Device Type, the test will report skipped for any sticky or default value portions of the test, when this type of Reset Sequence is used.)
3. If the function under test has an upstream port, the Secondary Bus Reset field in the Bridge Control register on the immediate downstream port (the port connected to the link of the function under test) is set to 1 using a WORD access, while preserving all the other fields in this register.
4. If the function under test is a Switch Downstream Port, the Secondary Bus Reset field in the Bridge Control register on the next higher downstream port (the port connected to the link of the Switch Upstream Port that the function under test is immediately subordinate to) is set to 1 using a WORD access, while preserving all the other fields in this register.
5. If the function under test is a PCI/PCI-X to PCI Express Bridge, the Secondary Bus Reset field in the Bridge Control register on the next higher bus device (the device sourcing the bus that the function under test is on) is set to 1 using a WORD access, while preserving all the other fields in this register.
6. If the function under test is a Root Port, the Secondary Bus Reset field in the Bridge Control register on the function under test is set to 1 using a WORD access, while preserving all the other fields in this register. (Note: For this Device Type, the test will report skipped for any sticky or default value portions of the test, when this type of Reset Sequence is used.)
7. After 50 ms the Secondary Bus Reset field that was previously set in any one of steps 3-6 is cleared to 0 using a WORD access, while preserving all the other fields in this register.
8. Software starts a timer which will be called the Reset Timeout Timer [RTT]. The [RTT] is started with a value of 0. Note: The [RTT] must be capable of counting up to at least 1 second.
9. (Note: In the interest of saving time, this step may be skipped if the function under test does not support 8.0 GT/s.) If the function under test has an upstream port, the immediate downstream port connected to the function under test is checked to see if it has completed training. If the function under test is a Switch Downstream Port, the immediate downstream port connected to the link of the Switch Upstream Port (that the function under test is immediately subordinate to) is checked to see if it has completed training. If the function under test has a non-Switch downstream port, the function under test is checked to see if it has completed training. Firstly the Data Link Layer Link Active Reporting Capable field in the Link Capabilities register is read and if it returns 0, then go on to the next step. If the Data Link Layer Link Active Reporting Capable field returned 1, then continuously read the Data Link Layer Link Active field in the Link Status register, and when it returns 1, go on to the next step.

If the Data Link Layer Link Active field does not return 1 before the [RTT] reaches 1 second, then report this as a link training failure, and skip the remaining steps.

10. Test software waits for 100 ms.
11. If the function under test is a Switch Downstream Port, then for the Switch Upstream Port (that the function under test is immediately subordinate to) test software restores all the values recorded in step 1 for: Primary Bus Number; Secondary Bus Number; Subordinate Bus Number.
12. (This step only applies to a function that is not an RCRB.) Read from the function under test, a WORD at location 00h (Vendor ID register) in Configuration Space and if it returns 0001h (CRS Software Visibility notification), then the read is repeated, until a different value is returned. If the different value is not returned before the [RTT] reaches 1 second, then report a function not detected failure, and skip the remaining steps.
13. (This step only applies to a function that is not a VF and not an RCRB.) Read from the function under test, a WORD at location 00h (Vendor ID register) in Configuration Space and if it returns FFFFh, then the read is repeated, until a different value is returned. If the different value is not returned before the [RTT] reaches 1 second, then report a function not detected failure, and skip the remaining steps.
14. If the function under test is a Switch Downstream Port, then steps 9-10 are repeated, by checking the downstream port of the function under test.
15. Stop the [RTT] timer.
16. Test software restores all the values recorded in step 1 for: Primary Bus Number; Secondary Bus Number; Subordinate Bus Number. These individual values must be restored in the Switch, all subordinate Switches and Bridges, and in the Type 1 Configuration Space and any subordinate Type 1 Configuration Spaces.

2.1.1.1.3 Function Level Reset (FLR) Reset Sequence

This is the default Reset Sequence selected automatically, for all functions under test that are Endpoints but are not connected to a link (e.g., Root Complex Integrated Endpoints). Since FLR does not affect the register fields that control the link of the device containing the function under test, it is not the primary choice for initializing functions under test that are connected to a link. When using FLR, test software must be aware of those register fields that are not reset (see Section 2.1.2.11). FLR is only supported on Endpoints. FLR is an optional feature and may not be implemented by the function under test.

1. If the function under test has a Type 1 Configuration Space header, then the following registers are read and their values recorded for later restoration: Primary Bus Number; Secondary Bus Number; Subordinate Bus Number. The values must be recorded for all of the following: a) if the function under test is not a Root Port then the function itself; b) if the function under test is a Switch Downstream Port then the Switch Upstream Port that the Function under test is subordinate to; c) all Switch Ports and Bridges connected below the link hierarchy originating from the function under test; d) all functions behind the Function under test that also have a Type 1 Configuration Space header. (Note: Resetting a Switch Downstream Port or PCI/PCI-X to PCI Express Bridge will cause it to transmit Hot Reset while resetting a PCI Express to PCI/PCI-X Bridge will cause it to assert bus reset, either of which will clear the bus number registers of all Switches and Bridges below it. This will make the Switch and Bridge functions

behind the Switch or Bridge disappear, so after the reset completes all these bus numbers must be restored before testing can continue.)

2. If the function under test has its Function Level Reset Capability field in the Device Capabilities register return 0, then this is not a failure and the remaining steps are skipped. (Note: The test will report skipped for any sticky or default value portions of the test.)
3. If the function under test has its Function Level Reset Capability field in the Device Capabilities register return 1, the Initiate Function Level Reset field on the function under test is set to 1 using a WORD access, while preserving all the other fields in this register.
4. Software starts a timer which will be called the Reset Timeout Timer [RTT]. The [RTT] is started with a value of 0. Note: The [RTT] must be capable of counting up to at least 1 second.
5. Test software waits for 100 ms.
6. (This step only applies to a function that is not an RCRB.) Read from the function under test, a WORD at location 00h (Vendor ID register) in Configuration Space and if it returns 0001h (CRS Software Visibility notification), then the read is repeated, until a different value is returned. If the different value is not returned before the [RTT] reaches 1 second, then report a function not detected failure, and skip the remaining steps.
7. (This step only applies to a function that is not a VF and not an RCRB.) Read from the function under test, a WORD at location 00h (Vendor ID register) in Configuration Space and if it returns FFFFh, then the read is repeated, until a different value is returned. If the different value is not returned before the [RTT] reaches 1 second, then report a function not detected failure, and skip the remaining steps.
8. Stop the [RTT] timer.
9. Test software restores all the values recorded in step 1 for: Primary Bus Number; Secondary Bus Number; Subordinate Bus Number. These individual values must be restored in the Switch, all subordinate Switches and Bridges, and in the Type 1 Configuration Space and any subordinate Type 1 Configuration Spaces.

2.1.1.2 D0-Uninitialized State

1. The appropriate Reset Sequence (described in Section 2.1.1.1) is executed, to put the function under test into its initial state.
2. Test software waits for 100 ms.
3. (This step only applies to a function under test that connects to a PCI Express link and if Base 2.x or later testing is being done.) If necessary, the Target Link Speed field is set to the target speed for the test (one of: 2.5 GT/s; 5.0 GT/s; 8.0 GT/s), per the steps described in Sections 2.1.2.12 to 2.1.2.14. (The test initiates the link speed change from the immediate downstream port connected to the function under test, but only if that downstream port supports selecting that link speed using the Target Link Speed field.)

Note that the packets (TLP/DLLP/Training Sets) from the downstream port may have any bits set that are newly defined in the latest *PCI Express Base Specification* but were reserved in previous revisions of the specification.

4. (This step only applies to a function under test that connects to a PCI Express link.) (In the interest of saving time, this step may be skipped if the function under test does not support 8.0 GT/s.) The immediate downstream port connected to the function under test is checked to see if it has completed training. Firstly the Data Link Layer Link Active Reporting Capable field in the Link Capabilities register is read and if it returns 0, then go on to the next step. If the Data Link Layer Link Active Reporting Capable field returned 1, then continuously read the Data Link Layer Link Active field in the Link Status register, and when it returns 1, go on to the next step. If the Data Link Layer Link Active field does not return 1 within 1 second, then report this as a link training failure, and skip the remaining steps.
5. Test software waits for 100 ms.
6. (This step only applies to a function that is not a VF and not an RCRB.) Read from the function under test, a WORD at location 00h (Vendor ID register) in Configuration Space and if it returns either 0001h (CRS Software Visibility Notification) or FFFFh, then the read is repeated, until a different value is returned. If the different value is not returned within 1 second, then report a function not detected failure, and skip the remaining steps.
7. (This step only applies to a function under test that connects to a PCI Express link and is not an RCRB.) Autonomous speed and bandwidth changes are disabled for all test cases unless explicitly noted. This is done by setting Hardware Autonomous Width Disable field to 1 in the Link Control register (if supported) and the Hardware Autonomous Speed Disable field to 1 in the Link Control 2 register (if supported). This is done on both upstream and downstream ports of the function under test's link. If the function under test is part of a multi-function device, then for the function under test's upstream port, this is done in function 0 of the function under test's device. If the function under test is a VF, then this is programmed in its associated PF.
8. At this point, the DUT has reached the D0-Uninitialized State. If this is the desired test state, then go to section to set the ASPM Settings. Otherwise continue with the steps in the appropriate Dx State section.

2.1.1.3 D0-Initialized State

This state begins with the DUT already programmed according to all the steps in the D0-Uninitialized State section.

1. (This step only applies to a function that is not a VF and not an RCRB.) Function BAR registers (if present) are configured to valid regions available in the test system.
Note: Bus Mastering, Interrupts, and PME are not enabled as part of the default configuration procedure.
2. (This step only applies to a function that is not a VF and not an RCRB.) Memory Space Enable and I/O Space Enable are set to 1 as appropriate, transitioning the function to D0-Initialized State.
3. At this point, the DUT has reached the D0-Initialized State. If this is the desired test state, then go to section to set the ASPM Settings. Otherwise continue with the steps in the Device State Settings section.

2.1.1.4 D1 State

This state begins with the DUT already programmed according to all the steps in the D0-Uninitialized State section.

1. For the function, read the D1 Support field (Power Management Capabilities register) and if it is 0, skip to step 4 (this is not a reported error).
2. For the function, the Power State field (Power Management Status and Control register) is set to 1.
3. For the function, the Power State field (Power Management Status and Control register) is polled repeatedly until it returns 1. If it does not return 1 within 1 second the polling stops (this is not a reported error).
4. Go to section to set the ASPM Settings.

2.1.1.5 D2 State

This state begins with the DUT already programmed according to all the steps in the D0-Uninitialized State section.

1. For the function, read the D2 Support field (Power Management Capabilities register) and if it is 0, skip to step 4 (this is not a reported error).
2. For the function, the Power State (Power Management Status and Control register) is set to 2.
3. For the function, the Power State (Power Management Status and Control register) is polled repeatedly until it returns 2. If it does not return 2 within 1 second the polling stops (this is not a reported error).
4. Go to section to set the ASPM Settings.

2.1.1.6 D3 (hot) State

This state begins with the DUT already programmed according to all the steps in the D0-Uninitialized State section.

1. For the function, the Power State (Power Management Status and Control register) is set to 3.
2. For the function, the Power State (Power Management Status and Control register) is polled repeatedly until it returns 3. If it does not return 3 within 1 second the polling stops (this is not a reported error).
3. Go to section to set the ASPM Settings.

2.1.1.7 Device State Settings

Device State: Some tests vary the starting power state of the device under test to be a non-D0 state. In those tests Device State is set at this stage in the initialization process. For a multi-function device under test, the setting is made in all functions of the device under test. Before these settings are made, the device under test is checked to ensure that the power state settings are supported by each function. If not supported by a function, the setting is skipped for that function only.

DUT: The Power State field (Power Management Control/Status register) is set as appropriate, for each supported function. Go to the section to set the ASPM Settings.

2.1.1.8 ASPM Settings

Active State Power Management (ASPM): Some tests vary the ASPM state of the upstream port of the device under test and the downstream port it is connected to. In those tests ASPM is set at this stage in the initialization process. For a multi-function device under test, the setting is made in all functions of the device under test. If the function under test is a VF, then this is programmed only in its associated PF. If the function under test is part of an internal link this is applied to both sides of the internal link. Before these settings are made, both the upstream port and the downstream port of the device under test's link are checked to ensure that the ASPM settings are supported. If not supported by either port, the setting is skipped.

Downstream Port: The Active State Power Management (ASPM) Control field (Link Control register for external links, Root Complex Link Control register for internal links) is set as appropriate.

Upstream Port: The Active State Power Management (ASPM) Control field (Link Control register for external links, Root Complex Link Control register for internal links) is set as appropriate.

2.1.2. Standard Register Field Characteristic Test Routines

The tests in this specification cover multiple PCI and PCI Express Configuration Space registers and capability structures. A common part of many of these tests is to check register field characteristics of each field in the register or capability structure under test. General procedures for testing all Configuration Space register field characteristics are given in this section. Individual test descriptions refer to this section.

The tests described below can be performed using accesses of different sizes. The smallest access that can be performed (read/write) is one byte (BYTE). Two byte (WORD) and four byte (DWORD) accesses can also be performed. The tests will use all accesses that can contain the full width of the field being tested. For example, if a 3 bit field is being tested, the test could use one, two, or four byte accesses in performing the test. If a 19 bit field is being tested, the test could use only four byte accesses in performing the test. The test can also use accesses that are not DWORD aligned as long as they do not cross a DWORD boundary and also that they contain the full width of the field being tested. For example, if the field being tested is bits 15 to 7, the test could use a four byte access aligned to byte 0, a two byte access aligned to byte 0, a two byte access aligned to byte 1, or a 1 byte access aligned to byte 1.

The following attribute options shall be supported by the test software: RO; RW; RW1C; ROS; RWS; RW1CS; RO-Zero; RO-Ones; HwInit. The specific test descriptions will mention which attribute options are used in testing each individual field.

Note: Some fields may be implemented as HwInit for Root Complex devices, and system-integrated devices. Test software that will execute on these devices before all system firmware has run on system startup must allow for these fields to be writable once (that is be HwInit).

2.1.2.1 Read Only (RO) Register Field Testing

1. The initial value is read from the field under test.
2. The inverse of the value read is written to the field under test. The field under test is inverted, while all other values are preserved if the register is a capabilities or a control register (RsvdP), or zeroed if the register is a status register (RsvdZ). No check is done to ensure that the write succeeds and no check is done to see if the request produced an error response from the function under test.
3. The field value is read. The value must be unchanged from the value read in step 1.
4. A binary value of all 1's is written to the field under test, while any other remaining field values are preserved if the register is a capabilities or a control register (RsvdP), or zeroed if the register is a status register (RsvdZ).
5. The field value is read. The value must be unchanged from the value read in step 1.

2.1.2.2 Read Write (RW) Register Field Testing

There are two types of RW register field testing. The first type of testing involves writing all valid values to each RW register field and verifying that the value written is read back. These cases are documented in individual test descriptions. The second type of testing involves writing illegal values to RW register fields and monitoring hardware behavior. These cases are documented in individual test descriptions. The following description describes the illegal value RW register procedure that is performed on all RW register fields.

1. The function is placed in the D0-Uninitialized state. (BARs are not configured).
 Note: This test only applies with the function is in the D0-Uninitialized state and its BAR registers have not been configured.
2. Any test-specific register field is restored to its described value in the test description (e.g., if the field under test is part of an indexed register, then the index is restored first before accessing the field under test).
3. The initial value is read from the field under test.
4. Any illegal value is written to the field under test, while any other remaining field values are preserved if the register is a capabilities or a control register (RsvdP), or zeroed if the register is a status register (RsvdZ).
 Note: The test does not check that the illegal value is latched by the field under test. This is not a requirement.
5. The original value read in step 3 is written to the device.
6. The function is configured to the D0-Initialized state following the procedure in Section 2.1.1.3. The configuration process must still work correctly.

2.1.2.3 Read Write 1 Clear (RW1C) Register Field Testing

1. The initial value is read from the field under test.
2. Test software writes all 0's to the field under test, while any other remaining field values are preserved if the register is a capabilities or a control register (RsvdP), or zeroed if the register is a status register (RsvdZ).
3. Test software reads the field under test. The value read must be unchanged from the value read in step 1.
4. Test software writes all 1's to the field under test while any other remaining field values are preserved if the register is a capabilities or a control register (RsvdP), or zeroed if the register is a status register (RsvdZ).
5. Test software reads the field under test. The value read must be all 0's.

2.1.2.4 Read Only Sticky (ROS) Register Field Testing

The tests in Section 2.1.2.1 for a RO register field are repeated.

The following additional tests are then performed:

1. The value is read from the field under test.
2. The appropriate Reset Sequence (described in Section 2.1.1.1) is executed, to put the function under test into its initial state.
3. Any test-specific register field is restored to its described value in the test description (e.g., if the field under test is part of an indexed register, then the index is restored first before accessing the field under test).
4. The field under test is read. The value must be unchanged from the value read in step 1.

Note: Special cases where values need to be maintained/cleared through the Reset Sequence are called out specifically in individual test descriptions later in this specification.

2.1.2.5 Read Write Sticky (RWS) Register Field Testing

The RW tests for a register field described in Section 2.1.2.2 are performed normally. The following additional testing is performed to verify the sticky nature of the register:

5. The value is read from the field under test.
6. Any legal value is written to the field under test, while any other remaining field values are preserved if the register is a capabilities or a control register (RsvdP), or zeroed if the register is a status register (RsvdZ).
7. The field under test is read. The value read must match the value written in step 2.
8. The appropriate Reset Sequence (described in Section 2.1.1.1) is executed, to put the function under test into its initial state.
9. Any test-specific register field is restored to its described value in the test description (e.g., if the field under test is part of an indexed register, then the index is restored first before accessing the field under test).
10. The field under test is read. The value must match the value read in step 3.

Note: Special cases where values need to be maintained/cleared through the Reset Sequence are called out specifically in individual test descriptions later in this specification.

2.1.2.6 Read Write 1 Clear Sticky (RW1CS) Register Field Testing

The following steps are performed in addition to the testing in Section 2.1.2.3 for a RW1C register:

1. The value is read from the field under test.
2. The appropriate Reset Sequence (described in Section 2.1.1.1) is executed, to put the function under test into its initial state.
3. Any test-specific register field is restored to its described value in the test description (e.g., if the field under test is part of an indexed register, then the index is restored first before accessing the field under test).
4. The field under test is read. It must match the value read in step 1.

Note: Special cases where values need to be maintained/cleared through the Reset Sequence are called out specifically in individual test descriptions later in this specification.

2.1.2.7 Hardware Initialized (HwInit) Register Field Testing

1. The value is read from the field under test.
2. The inverse of the value read and if necessary converted to a legal value and then is written to the field under test. The field under test is inverted while all other values are preserved if the register is a capabilities or a control register (RsvdP), or zeroed if the register is a status register (RsvdZ). No check is done to ensure that the write succeeds and no check is done to see if the request produced an error response from the function under test.
3. The value of the field under test is read. If the value is unchanged from the value read in step 1, continue to the next step. If the value is changed then steps 1-3 are repeated just once more. The value is only allowed to change once, if it changes again, then the test fails.
4. A binary value of all 1's is written to the field under test, while any other remaining field values are preserved if the register is a capabilities or a control register (RsvdP), or zeroed if the register is a status register (RsvdZ).
5. The field under test is read. The value must be unchanged from the value read in step 3. If the value has changed the test fails.

2.1.2.8 RO-Zero Register Field Testing

RO-Zero register fields are tested with the same requirements as Read Only (RO) register fields except that every read must return all bits in the field under test as 0. The software requirement to preserve the existing bits in an RsvdP field is ignored by test software. The software requirement to only write 0's to an RsvdZ field is ignored by test software.

2.1.2.9 RO-Ones Register Field Testing

RO-Ones register fields are tested with the same requirements as Read Only register fields except that every read must return all bits in the field under test as 1.

2.1.2.10 Default Value Register Field Testing

Default value register field testing can only be executed on those device types that support a Reset Sequence that can be used to return the function under test to its default state. Devices with only a Downstream port (a Root Port or a PCI/PCI-X to PCI Express Bridge), cannot be reset by any of the Reset Sequences defined in this test document. Therefore, any function under test that is a Root Port or a PCI/PCI-X to PCI Express Bridge will skip the default value register field part of the test descriptions. Root Complex internal components (a Root Complex Integrated Endpoint, a Root Complex Event Collector, or an RCRB), similarly cannot be reset by any of the Reset Sequences defined in this test document, except in the case that they support Function Level Reset. Therefore, any function under test that is a Root Complex Integrated Endpoint without FLR support, a Root Complex Event Collector, or an RCRB will skip the default value register field part of the test descriptions.

Base specification rules for default value register fields do not apply to Root Complexes and system-integrated devices. As such these devices may return any default value. Test software has no way to determine if the function under test is a system-integrated device, so it treats all functions under test as if they were not system-integrated devices. For system-integrated devices other than those listed in the preceding two paragraphs, the test results for default value register field testing, may require the user to manually review the results in order to determine if the default value register field failure is valid.

1. If the field under test is RW1C or RW1CS, the default value listed in the test description is written to the field under test, while any other remaining field values are preserved if the register is a capabilities or a control register (RsvdP), or zeroed if the register is a status register (RsvdZ). If the field under test is not RW1C and not RW1CS, the inverse of the default value listed in the test description is written to the field under test, while any other remaining field values are preserved if the register is a capabilities or a control register (RsvdP), or zeroed if the register is a status register (RsvdZ). No check is done to ensure that the write succeeds and no check is done to see if the request produced an error response from the function under test. (Note: Since a RW1C or RW1CS field is cleared by writing a 1, and it is not changed by writing a 0, it behaves in an inverse way to a RW or RWS field. Therefore, to test the default value for a RW1C or RW1CS field, the value written must be one that tries to have the field contain the opposite value specified as its default value. Examples: If the field is RW1C defaulting to 0 and it currently has a 1 in it, then to test for default value, write it with a 0, so it retains the 1, and then initiate the Reset Sequence to cause the field to reset to 0. If the field is RW1C defaulting to 1 and it currently has a 1 in it, then to test for default value, write it with a 1, so it clears to 0, and then initiate the Reset Sequence to cause the field to reset to 1.)
2. The value in the field under test is read.
3. The appropriate Reset Sequence (described in Section 2.1.1.1) is executed, to put the function under test into its initial state.
4. Any test-specific register field is restored to its described value in the test description (e.g., if the field under test is part of an indexed register, then the index is restored first before accessing the field under test).
5. The value in the field under test is read.
 - a. For a sticky field, it must match the value read in step 2.
 - b. For a non-sticky field, it must match the default value listed in the test description for that field.

2.1.2.11 Function Level Reset Testing

For a function or PF that supports Function Level Reset (as determined by checking the Function Level Reset Capability field in the Device Control register) all register fields default value tests that did not use a FLR Reset Sequence are repeated using a FLR Reset Sequence (described in Section 2.1.1.1.3).

For this specific FLR testing, all fields described as HwInit, must be tested as RO instead.

For this specific FLR testing the following fields must not be tested for the default values listed in the test description:

Device Capabilities Register

- a. Captured Slot Power Limit Value
- b. Captured Slot Power Limit Scale

Device Control Register

- a. Max_Payload_Size

Link Control Register

- a. Active State Power Management (ASPM) Control
- b. Read Completion Boundary (RCB)
- c. Common Clock Configuration
- d. Extended Synch
- e. Enable Clock Power Management
- f. Hardware Autonomous Width Disable

Link Control 2 Register

- a. Hardware Autonomous Speed Disable

Virtual Channel Capability Structure

- a. All registers

Multi Function Virtual Channel Capability Structure

- a. All registers

Secondary PCI Express Capability Structure

- a. All Equalization Control registers

Note: Sticky fields are never tested for default values when FLR is used as the Reset Sequence.

For this specific FLR testing, all other fields not listed above are tested for register fields default value according to the requirements of the test description.

2.1.2.12 2.5 GT/s Support Detection and Testing

The following applies to Base 1.x or later testing. All links are assumed to support 2.5 GT/s and only this link speed will be tested. No specific algorithm is needed to detect the supported link speed, and no attempt is made to change the current link speed. The test assumes that the hardware will automatically train the link to 2.5 GT/s, if that is the only supported link speed, whenever the link is enabled or reset.

The following only applies to Base 2.x or later testing.

For a function under test that supports greater than 2.5 GT/s, test software will need to limit the link speed to 2.5 GT/s using the Target Link Speed field in the Link Control 2 register, in order to test at this speed.

In order to configure the function under test's external PCI Express link to 2.5 GT/s the following steps are done if the function under test contains an upstream port (e.g., Endpoint, Switch Upstream Port, PCI Express to PCI/PCI-X Bridge):

1. Read the function under test's Link Capabilities register bits 3-0 value and if it returns 0000b (no link supported) or 0001b (only 2.5 GT/s supported), then no further attempt is made to change the link speed to 2.5 GT/s.
2. Read the immediate downstream port's (the port connected to the function under test) Link Capabilities register bits 3-0 value and if it returns 0000b (no link supported) or 0001b (only 2.5 GT/s supported), then no further attempt is made to change the link speed to 2.5 GT/s.
3. If the Capability Version field (PCI Express Capabilities register) is 2h or greater: the Target Link Speed field (Link Control 2 register) is set to 0001b (2.5 GT/s) on function 0 of the device containing the function under test. For this write the following fields are also set as follows:
 - a. Enter Compliance (bit 4) is set to 0
 - b. Hardware Autonomous Speed Disable (bit 5) is set to 1
 - c. Transmit Margin (bits 9-7) is set to 000b
 - d. Enter Modified Compliance (bit 10) is set to 0
 - e. Compliance SOS (bit 11) is set to 0
 - f. Compliance Preset/De-emphasis (bits 15-12) is set to 0000b.
4. If the Capability Version field (PCI Express Capabilities register) is 2h or greater: the Target Link Speed field (Link Control 2 register) is read back and if it does not return 0001b (2.5 GT/s) or 0000b (reserved) on function 0 of the device containing the function under test then no further attempt is made to change the link speed to 2.5 GT/s. This is reported as a test failure.
5. If the Capability Version field (PCI Express Capabilities register) is 2h or greater: the Target Link Speed field (Link Control 2 register) is set to 0001b (2.5 GT/s) on the immediate downstream port (the port connected to the function under test). This port will be referred to as the downstream port of the link. For this write the following fields are also set as follows:
 - a. Enter Compliance (bit 4) is set to 0
 - b. Hardware Autonomous Speed Disable (bit 5) is set to 1
 - c. Transmit Margin (bits 9-7) is set to 000b
 - d. Enter Modified Compliance (bit 10) is set to 0
 - e. Compliance SOS (bit 11) is set to 0
 - f. Compliance Preset/De-emphasis (bits 15-12) is set to 0000b.
6. If the Capability Version field (PCI Express Capabilities register) is 2h or greater: the Target Link Speed field (Link Control 2 register) is read back and if it does not return 0001b (2.5 GT/s) or 0000b (reserved) on the downstream port of the link, then no further attempt is made to change the link speed to 2.5 GT/s (this is not a failure, but the test result for 2.5 GT/s is reported as skipped).

7. Read the downstream port's Link Capabilities register bits 3-0 value and if it returns 0011b (8.0 GT/s supported) or greater, then:
 - a. The Perform Equalization field (Link Control 3 register) is set to 0 on the downstream port of the link.
8. Test software reads the Link Training field (Link Status register) on the downstream port of the link until it reads 0. If the Link Training field does not return 0 within 1 second, then report this as a link training failure, and skip the remaining steps. (Note: This step is needed to ensure that the next retrain will use the settings programmed by the preceding steps.)
9. Test software writes 1 to the Retrain Link field (Link Control register) on the downstream port of the link using a WORD access, while preserving all the other fields in this register.
10. Test software reads the Link Training field (Link Status register) on the downstream port of the link until it reads 0. If the Link Training field does not return 0 within 1 second, then report this as a link training failure, and skip the remaining steps.
11. Test software reads the Current Link Speed field (Link Status register) in the downstream port of the link. The value must return 0001b (2.5 GT/s). If it does not, then the attempt fails and is reported as a link speed failure.

In order to configure the function under test's external PCI Express link to 2.5 GT/s the following steps are done if the function under test contains a downstream port (e.g., Root Port, Switch Downstream Port, PCI/PCI-X to PCI Express Bridge):

1. Read the function under test's Link Capabilities register bits 3-0 value and if it returns 0000b (no link supported) or 0001b (only 2.5 GT/s supported), then no further attempt is made to change the link speed to 2.5 GT/s.
2. Read the immediate upstream port's (connected to the function under test) Link Capabilities register bits 3-0 value and if it returns 0000b (no link supported) or 0001b (only 2.5 GT/s supported), then no further attempt is made to change the link speed to 2.5 GT/s.
3. If the Capability Version field (PCI Express Capabilities register) is 2h or greater: the Target Link Speed field (Link Control 2 register) is set to 0001b (2.5 GT/s) on function 0 of the immediate upstream port (connected to the function under test). For this write the following fields are also set as follows:
 - a. Enter Compliance (bit 4) is set to 0
 - b. Hardware Autonomous Speed Disable (bit 5) is set to 1
 - c. Transmit Margin (bits 9-7) is set to 000b
 - d. Enter Modified Compliance (bit 10) is set to 0
 - e. Compliance SOS (bit 11) is set to 0
 - f. Compliance Preset/De-emphasis (bits 15-12) is set to 0000b.
4. If the Capability Version field (PCI Express Capabilities register) is 2h or greater: the Target Link Speed field (Link Control 2 register) is read back and if it does not return 0001b (2.5 GT/s) or 0000b (reserved) on function 0 of the immediate upstream port (connected to the function under test) then no further attempt is made to change the link speed to 2.5 GT/s (this is not a failure, but the test result for 2.5 GT/s is reported as skipped).

5. If the Capability Version field (PCI Express Capabilities register) is 2h or greater: the Target Link Speed field (Link Control 2 register) is set to 0001b (2.5 GT/s) on the function under test. For this write the following fields are also set as follows:
 - a. Enter Compliance (bit 4) is set to 0
 - b. Hardware Autonomous Speed Disable (bit 5) is set to 1
 - c. Transmit Margin (bits 9-7) is set to 000b
 - d. Enter Modified Compliance (bit 10) is set to 0
 - e. Compliance SOS (bit 11) is set to 0
 - f. Compliance Preset/De-emphasis (bits 15-12) is set to 0000b.
6. If the Capability Version field (PCI Express Capabilities register) is 2h or greater: the Target Link Speed field (Link Control 2 register) is read back and if it does not return 0001b (2.5 GT/s) or 0000b (reserved) on the function under test then no further attempt is made to change the link speed to 2.5 GT/s. This is reported as a test failure.
7. Read the function under test's Link Capabilities register bits 3-0 value and if it returns 0011b (8.0 GT/s supported) or greater, then:
 - a. The Perform Equalization field (Link Control 3 register) is set to 0 on the function under test.
8. Test software reads the Link Training field (Link Status register) on the function under test until it reads 0. If the Link Training field does not return 0 within 1 second, then report this as a link training failure, and skip the remaining steps. (Note: This step is needed to ensure that the next retrain will use the settings programmed by the preceding steps.)
9. Test software writes 1 to the Retrain Link field (Link Control register) on the function under test using a WORD access, while preserving all the other fields in this register.
10. Test software reads the Link Training field (Link Status register) on the function under test until it reads 0. If the Link Training field does not return 0 within 1 second, then report this as a link training failure, and skip the remaining steps.
11. Test software reads the Current Link Speed field (Link Status register) in the function under test and it must return 0001b (2.5 GT/s). If it does not, then the attempt fails and is reported as a link speed failure.

Test software cannot control the link speed of Root Complex internal device links (Root Complex Integrated Endpoints or Root Complex Event Collectors). When a function under test is of this device type, no attempt is made to control the link speed and the test is run only once at whatever link speed the internal link runs at. In addition, some Root Ports may connect to internal links (rather than external links). Test software will only try to change the link speed on such a Root Port's external link (if it has one). It is not currently possible to configure the speed of Root Complex internal links, so there is currently no support in test software to do so.

2.1.2.13 5.0 GT/s Support Detection and Testing

The following only applies to Base 2.x or later testing. The test assumes that all links support at least 2.5 GT/s.

For a function under test that supports 5.0 GT/s, test software will attempt to test it once at 2.5 GT/s (which all PCI Express devices with external links, must support), and again at 5.0 GT/s (which is an optional feature starting with Base 2.0).

For a function under test that supports greater than 5.0 GT/s, test software will need to limit the link speed to 5.0 GT/s using the Target Link Speed field in the Link Control 2 register, in order to test at this speed.

In order to configure the function under test's external PCI Express link to 5.0 GT/s the following steps are done if the function under test contains an upstream port (Endpoint, Switch Upstream Port, or PCI Express to PCI/PCI-X Bridge):

1. Read the function under test's Link Capabilities register bits 3-0 value and if it returns 0000b (no link supported) or 0001b (only 2.5 GT/s supported), then no further attempt is made to change the link speed to 5.0 GT/s.
2. Read the immediate downstream port's (the port connected to the function under test) Link Capabilities register bits 3-0 value and if it returns 0000b (no link supported) or 0001b (only 2.5 GT/s supported), then no further attempt is made to change the link speed to 5.0 GT/s.
3. If the Capability Version field (PCI Express Capabilities register) is 2h or greater: the Target Link Speed field (Link Control 2 register) is set to 0010b (5.0 GT/s) on function 0 of the device containing the function under test. For this write the following fields are also set as follows:
 - a. Enter Compliance (bit 4) is set to 0
 - b. Hardware Autonomous Speed Disable (bit 5) is set to 1
 - c. Transmit Margin (bits 9-7) is set to 000b
 - d. Enter Modified Compliance (bit 10) is set to 0
 - e. Compliance SOS (bit 11) is set to 0
 - f. Compliance Preset/De-emphasis (bits 15-12) is set to 0000b.
4. If the Capability Version field (PCI Express Capabilities register) is 2h or greater: the Target Link Speed field (Link Control 2 register) is read back and if it does not return 0010b (5.0 GT/s) on function 0 of the device containing the function under test then no further attempt is made to change the link speed to 5.0 GT/s. This is reported as a test failure.
5. If the Capability Version field (PCI Express Capabilities register) is 2h or greater: the Target Link Speed field (Link Control 2 register) is set to 0010b (5.0 GT/s) on the immediate downstream port (the port connected to the function under test). This port will be referred to as the downstream port of the link. For this write the following fields are also set as follows:
 - a. Enter Compliance (bit 4) is set to 0
 - b. Hardware Autonomous Speed Disable (bit 5) is set to 1
 - c. Transmit Margin (bits 9-7) is set to 000b
 - d. Enter Modified Compliance (bit 10) is set to 0
 - e. Compliance SOS (bit 11) is set to 0
 - f. Compliance Preset/De-emphasis (bits 15-12) is set to 0000b.
6. If the Capability Version field (PCI Express Capabilities register) is 2h or greater: the Target Link Speed field (Link Control 2 register) is read back and if it does not return 0010b (5.0 GT/s) on the downstream port of the link, then no further attempt is made to change the link speed to 5.0 GT/s (this is not a failure, but the test result for 5.0 GT/s is reported as skipped).
7. Read the downstream port's Link Capabilities register bits 3-0 value and if it returns 0011b (8.0 GT/s supported) or greater, then:
 - a. The Perform Equalization field (Link Control 3 register) is set to 0 on the downstream port of the link.

8. Test software reads the Link Training field (Link Status register) on the downstream port of the link until it reads 0. If the Link Training field does not return 0 within 1 second, then report this as a link training failure, and skip the remaining steps. (Note: This step is needed to ensure that the next retrain will use the settings programmed by the preceding steps.)
9. Test software writes 1 to the Retrain Link field (Link Control register) on the downstream port of the link using a WORD access, while preserving all the other fields in this register.
10. Test software reads the Link Training field (Link Status register) on the downstream port of the link until it reads 0. If the Link Training field does not return 0 within 1 second, then report this as a link training failure, and skip the remaining steps.
11. Test software reads the Current Link Speed field (Link Status register) in the downstream port of the link. If the value returns 0010b (5.0 GT/s), then skip the remaining steps.
12. Test software writes 1 to the Retrain Link field (Link Control register) on the downstream port of the link using a WORD access, while preserving all the other fields in this register.
13. Test software reads the Link Training field (Link Status register) on the downstream port of the link until it reads 0. If the Link Training field does not return 0 within 1 second, then report this as a link training failure, and skip the remaining steps.
14. Test software reads the Current Link Speed field (Link Status register) in the downstream port of the link. The value must return 0010b (5.0 GT/s). If it does not, then the attempt fails and is reported as a link speed failure.

In order to configure the function under test's external PCI Express link to 5.0 GT/s the following steps are done if the function under test contains a downstream port (Root Port, Switch Downstream Port, or PCI/PCI-X to PCI Express Bridge):

1. Read the function under test's Link Capabilities register bits 3-0 value and if it returns 0000b (no link supported) or 0001b (only 2.5 GT/s supported), then no further attempt is made to change the link speed to 5.0 GT/s.
2. Read the immediate upstream port's (connected to the function under test) Link Capabilities register bits 3-0 value and if it returns 0000b (no link supported) or 0001b (only 2.5 GT/s supported), then no further attempt is made to change the link speed to 5.0 GT/s.
3. If the Capability Version field (PCI Express Capabilities register) is 2h or greater: the Target Link Speed field (Link Control 2 register) is set to 0010b (5.0 GT/s) on function 0 of the immediate upstream port (connected to the function under test). For this write the following fields are also set as follows:
 - a. Enter Compliance (bit 4) is set to 0
 - b. Hardware Autonomous Speed Disable (bit 5) is set to 1
 - c. Transmit Margin (bits 9-7) is set to 000b
 - d. Enter Modified Compliance (bit 10) is set to 0
 - e. Compliance SOS (bit 11) is set to 0
 - f. Compliance Preset/De-emphasis (bits 15-12) is set to 0000b.
4. If the Capability Version field (PCI Express Capabilities register) is 2h or greater: the Target Link Speed field (Link Control 2 register) is read back and if it does not return 0010b (5.0 GT/s) on function 0 of the immediate upstream port (connected to the function under test) then no further attempt is made to change the link speed to 5.0 GT/s (this is not a failure, but the test result for 5.0 GT/s is reported as skipped).

5. If the Capability Version field (PCI Express Capabilities register) is 2h or greater: the Target Link Speed field (Link Control 2 register) is set to 0010b (5.0 GT/s) on the function under test. For this write the following fields are also set as follows:
 - a. Enter Compliance (bit 4) is set to 0
 - b. Hardware Autonomous Speed Disable (bit 5) is set to 1
 - c. Transmit Margin (bits 9-7) is set to 000b
 - d. Enter Modified Compliance (bit 10) is set to 0
 - e. Compliance SOS (bit 11) is set to 0
 - f. Compliance Preset/De-emphasis (bits 15-12) is set to 0000b.
6. If the Capability Version field (PCI Express Capabilities register) is 2h or greater: the Target Link Speed field (Link Control 2 register) is read back and if it does not return 0010b (5.0 GT/s) on the function under test then no further attempt is made to change the link speed to 5.0 GT/s. This is reported as a test failure.
7. Read the function under test's Link Capabilities register bits 3-0 value and if it returns 0011b (8.0 GT/s supported) or greater, then:
 - a. The Perform Equalization field (Link Control 3 register) is set to 0 on the function under test.
8. Test software reads the Link Training field (Link Status register) on the function under test until it reads 0. If the Link Training field does not return 0 within 1 second, then report this as a link training failure, and skip the remaining steps. (Note: This step is needed to ensure that the next retrain will use the settings programmed by the preceding steps.)
9. Test software writes 1 to the Retrain Link field (Link Control register) on the function under test using a WORD access, while preserving all the other fields in this register.
10. Test software reads the Link Training field (Link Status register) on the function under test until it reads 0. If the Link Training field does not return 0 within 1 second, then report this as a link training failure, and skip the remaining steps.
11. Test software reads the Current Link Speed field (Link Status register) in the function under test and if it returns 0010b (5.0 GT/s), then skip the remaining steps.
12. Test software writes 1 to the Retrain Link field (Link Control register) on the downstream port of the link using a WORD access, while preserving all the other fields in this register.
13. Test software reads the Link Training field (Link Status register) on the downstream port of the link until it reads 0. If the Link Training field does not return 0 within 1 second, then report this as a link training failure, and skip the remaining steps.
14. Test software reads the Current Link Speed field (Link Status register) in the function under test and it must return 0010b (5.0 GT/s). If it does not, then the attempt fails and is reported as a link speed failure.

Test software cannot control the link speed of Root Complex internal device links (Root Complex Integrated Endpoints or Root Complex Event Collectors). When a function under test is of this device type, no attempt is made to control the link speed and the test is run only once at whatever link speed the internal link runs at. In addition, some Root Ports may connect to internal links (rather than external links). Test software will only try to change the link speed on such a Root Port's external link (if it has one). It is not currently possible to configure the speed of Root Complex internal links, so there is currently no support in test software to do so.

2.1.2.14 8.0 GT/s Support Detection and Testing

The following only applies to Base 3.x or later testing. The test assumes that all links support at least 2.5 GT/s.

For a function under test that supports 8.0 GT/s, test software will attempt to test it once at 2.5 GT/s (which all PCI Express devices with external links, must support), again at 5.0 GT/s (which is an optional feature starting with Base 2.0), and again at 8.0 GT/s (which is an optional feature starting with Base 3.0).

For a function under test that supports greater than 8.0 GT/s, test software will need to limit the link speed to 8.0 GT/s using the Target Link Speed field in the Link Control 2 register, in order to test at this speed.

In order to configure the function under test's external PCI Express link to 8.0 GT/s the following steps are done if the function under test contains an upstream port (Endpoint, Switch Upstream Port, or PCI Express to PCI/PCI-X Bridge):

1. Read the function under test's Link Capabilities register bits 3-0 value and if it returns 0000b (no link supported), or 0001b (only 2.5 GT/s supported), or 0010b (only 2.5 GT/s and 5.0 GT/s supported), then no further attempt is made to change the link speed to 8.0 GT/s.
2. Read the immediate downstream port's (the port connected to the function under test) Link Capabilities register bits 3-0 value and if it returns 0000b (no link supported), or 0001b (only 2.5 GT/s supported), or 0010b (only 2.5 GT/s and 5.0 GT/s supported), then no further attempt is made to change the link speed to 8.0 GT/s.
3. If the Capability Version field (PCI Express Capabilities register) is 2h or greater: the Target Link Speed field (Link Control 2 register) is set to 0011b (8.0 GT/s) on function 0 of the device containing the function under test. For this write the following fields are also set as follows:
 - a. Enter Compliance (bit 4) is set to 0
 - b. Hardware Autonomous Speed Disable (bit 5) is set to 1
 - c. Transmit Margin (bits 9-7) is set to 000b
 - d. Enter Modified Compliance (bit 10) is set to 0
 - e. Compliance SOS (bit 11) is set to 0
 - f. Compliance Preset/De-emphasis (bits 15-12) is set to 0000b.
4. If the Capability Version field (PCI Express Capabilities register) is 2h or greater: the Target Link Speed field (Link Control 2 register) is read back and if it does not return 0011b (8.0 GT/s) on function 0 of the device containing the function under test then no further attempt is made to change the link speed to 8.0 GT/s. This is reported as a test failure.
5. If the Capability Version field (PCI Express Capabilities register) is 2h or greater: the Target Link Speed field (Link Control 2 register) is set to 0011b (8.0 GT/s) on the immediate downstream port (the port connected to the function under test). This port will be referred to as the downstream port of the link. For this write the following fields are also set as follows:
 - a. Enter Compliance (bit 4) is set to 0
 - b. Hardware Autonomous Speed Disable (bit 5) is set to 1
 - c. Transmit Margin (bits 9-7) is set to 000b
 - d. Enter Modified Compliance (bit 10) is set to 0
 - e. Compliance SOS (bit 11) is set to 0
 - f. Compliance Preset/De-emphasis (bits 15-12) is set to 0000b.

6. If the Capability Version field (PCI Express Capabilities register) is 2h or greater: the Target Link Speed field (Link Control 2 register) is read back and if it does not return 0011b (8.0 GT/s) on the downstream port of the link, then no further attempt is made to change the link speed to 8.0 GT/s (this is not a failure, but the test result for 8.0 GT/s is reported as skipped).
7. The Perform Equalization field (Link Control 3 register) is set to 0 on the downstream port of the link.
8. Test software reads the Link Training field (Link Status register) on the downstream port of the link until it reads 0. If the Link Training field does not return 0 within 1 second, then report this as a link training failure, and skip the remaining steps. (Note: This step is needed to ensure that the next retrain will use the settings programmed by the preceding steps.)
9. Test software writes 1 to the Retrain Link field (Link Control register) on the downstream port of the link using a WORD access, while preserving all the other fields in this register.
10. Test software reads the Link Training field (Link Status register) on the downstream port of the link until it reads 0. If the Link Training field does not return 0 within 1 second, then report this as a link training failure, and skip the remaining steps.
11. Test software reads the Current Link Speed field (Link Status register) in the downstream port of the link. If the value returns 0011b (8.0 GT/s), then skip the remaining steps.
12. Test software writes 1 to the Retrain Link field (Link Control register) on the downstream port of the link using a WORD access, while preserving all the other fields in this register.
13. Test software reads the Link Training field (Link Status register) on the downstream port of the link until it reads 0. If the Link Training field does not return 0 within 1 second, then report this as a link training failure, and skip the remaining steps.
14. Test software reads the Current Link Speed field (Link Status register) in the downstream port of the link. The value must return 0011b (8.0 GT/s). If it does not, then the attempt fails and is reported as a link speed failure.

In order to configure the function under test's external PCI Express link to 8.0 GT/s the following steps are done if the function under test contains a downstream port (Root Port, Switch Downstream Port, or PCI/PCI-X to PCI Express Bridge):

1. Read the function under test's Link Capabilities register bits 3-0 value and if it returns 0000b (no link supported), or 0001b (only 2.5 GT/s supported), or 0010b (only 2.5 GT/s and 5.0 GT/s supported), then no further attempt is made to change the link speed to 8.0 GT/s.
2. Read the immediate upstream port's (connected to the function under test) Link Capabilities register bits 3-0 value and if it returns 0000b (no link supported), or 0001b (only 2.5 GT/s supported), or 0010b (only 2.5 GT/s and 5.0 GT/s supported), then no further attempt is made to change the link speed to 8.0 GT/s.
3. If the Capability Version field (PCI Express Capabilities register) is 2h or greater: the Target Link Speed field (Link Control 2 register) is set to 0011b (8.0 GT/s) on function 0 of the immediate upstream port (connected to the function under test). For this write the following fields are also set as follows:
 - a. Enter Compliance (bit 4) is set to 0
 - b. Hardware Autonomous Speed Disable (bit 5) is set to 1
 - c. Transmit Margin (bits 9-7) is set to 000b

- d. Enter Modified Compliance (bit 10) is set to 0
 - e. Compliance SOS (bit 11) is set to 0
 - f. Compliance Preset/De-emphasis (bits 15-12) is set to 0000b.
4. If the Capability Version field (PCI Express Capabilities register) is 2h or greater: the Target Link Speed field (Link Control 2 register) is read back and if it does not return 0011b (8.0 GT/s) on function 0 of the immediate upstream port (connected to the function under test) then no further attempt is made to change the link speed to 8.0 GT/s (this is not a failure, but the test result for 8.0 GT/s is reported as skipped).
 5. If the Capability Version field (PCI Express Capabilities register) is 2h or greater: the Target Link Speed field (Link Control 2 register) is set to 0011b (8.0 GT/s) on the function under test. For this write the following fields are also set as follows:
 - a. Enter Compliance (bit 4) is set to 0
 - b. Hardware Autonomous Speed Disable (bit 5) is set to 1
 - c. Transmit Margin (bits 9-7) is set to 000b
 - d. Enter Modified Compliance (bit 10) is set to 0
 - e. Compliance SOS (bit 11) is set to 0
 - f. Compliance Preset/De-emphasis (bits 15-12) is set to 0000b.
 6. If the Capability Version field (PCI Express Capabilities register) is 2h or greater: the Target Link Speed field (Link Control 2 register) is read back and if it does not return 0011b (8.0 GT/s) on the function under test then no further attempt is made to change the link speed to 8.0 GT/s. This is reported as a test failure.
 7. The Perform Equalization field (Link Control 3 register) is set to 0 on the function under test.
 8. Test software reads the Link Training field (Link Status register) on the function under test until it reads 0. If the Link Training field does not return 0 within 1 second, then report this as a link training failure, and skip the remaining steps. (Note: This step is needed to ensure that the next retrain will use the settings programmed by the preceding steps.)
 9. Test software writes 1 to the Retrain Link field (Link Control register) on the function under test using a WORD access, while preserving all the other fields in this register.
 10. Test software reads the Link Training field (Link Status register) on the function under test until it reads 0. If the Link Training field does not return 0 within 1 second, then report this as a link training failure, and skip the remaining steps.
 11. Test software reads the Current Link Speed field (Link Status register) in the function under test and if it returns 0011b (8.0 GT/s), then skip the remaining steps.
 12. Test software writes 1 to the Retrain Link field (Link Control register) on the function under test using a WORD access, while preserving all the other fields in this register.
 13. Test software reads the Link Training field (Link Status register) on the function under test until it reads 0. If the Link Training field does not return 0 within 1 second, then report this as a link training failure, and skip the remaining steps.
 14. Test software reads the Current Link Speed field (Link Status register) in the function under test and it must return 0011b (8.0 GT/s). If it does not, then the attempt fails and is reported as a link speed failure.

Note: For the purposes of setting the link speed, test software assumes that for 8.0 GT/s, Link Equalization has already been performed (following reset of the function under test).

Test software cannot control the link speed of Root Complex internal device links (Root Complex Integrated Endpoints or Root Complex Event Collectors). When a function under test is of this device type, no attempt is made to control the link speed and the test is run only once at whatever link speed the internal link runs at. In addition, some Root Ports may connect to internal links (rather than external links). Test software must only try to change the link speed on such a Root Port's external link (if it has one). It is not currently possible to configure the speed of Root Complex internal links, so there is currently no support in test software to do so.

2.1.2.15 Capability Header Detection

Each implemented function will respond to Configuration Space accesses for addresses 00h to FFh. These functions may optionally implement a Capabilities List containing specific Capability structures. To detect the presence of a specific Capability structure the following procedure is done.

1. Set the number of instances value [CAPIN] to 0.
2. Read back a WORD at location 06h (Status register) and if bit 4 is 0, then skip to step 8 as this Bus Number/Device Number/Function Number location does not contain any Capability structures.
3. Read back a byte at location 34h (Capabilities Pointer register) and if 00h is found, then skip to step 8 as this Bus Number/Device Number/Function Number location does not contain any Capability structures.
4. Read back a WORD at the address location contained in Capabilities Pointer and record the value in [CAPHDR].
5. For [CAPHDR], if bits 7-0 (Capability ID) match the requested Capability ID, increment [CAPIN] by 1, and set the Capability base address value array [CAPBASE(n)] (where n is the value in [CAPIN]) to the address of the last read of [CAPHDR].
6. For [CAPHDR], if bits 15-8 (Next Capability Pointer) are 00h, then skip to step 8 as this Bus Number/Device Number/Function Number location does not contain any more Capability structures.
7. For [CAPHDR], if bits 15-8 are non-zero, read back a WORD at the address location contained in [CAPHDR] bits 15-8 and record the new value in [CAPHDR]. Repeat steps 4-7. There must be a limit of 64 repetitions of this step per function, so that a function that incorrectly implements an infinite loop of Capabilities Pointers will not cause the test to hang. If this limit is reached skip to step 8.
8. Return the Capability base address value array [CAPBASE (n)]. This is the starting address of each instance of the requested Capability ID. Return the number of instances value [CAPIN].

2.1.2.16 Extended Capability Header Detection

Each implemented PCI Express function will respond to Configuration Space accesses for addresses 100h to FFFh. These functions may optionally implement an Extended Capabilities List containing specific Extended Capability structures. To detect the presence of a specific Extended Capability structure the following procedure is done.

1. Set the number of instances value [ECAPIN] to 0.
2. Read back a DWORD at the address location 100h and record the value in [ECAPHDR].
3. For [ECAPHDR], if bits 15-0 (Extended Capability ID) match the requested Extended Capability ID, increment [ECAPIN] by 1, and set the Extended Capability base address value array [ECAPBASE (n)] (where n is the value in [ECAPIN]) to the address of the last read of [ECAPHDR].
4. For [ECAPHDR], if bits 31-20 (Next Capability Offset) are 000h, then skip to step 6 as this Bus Number/Device Number/Function Number location does not contain any more Extended Capability structures.
5. For [ECAPHDR], if bits 31-20 are non-zero, read back a DWORD at the address location contained in [ECAPHDR] bits 31-20 and record the new value in [ECAPHDR]. Repeat steps 3-5. There must be a limit of 1024 repetitions of this step per function, so that a function that incorrectly implements an infinite loop of Extended Capabilities Pointers will not cause the test to hang. If this limit is reached skip to step 6.
6. Return the Extended Capability base address value array [ECAPBASE (n)]. This is the starting address of each instance of the requested Extended Capability ID. Return the number of instances value [ECAPIN].

2.1.2.17 Trusted Configuration Space Header Detection

Support for Trusted Configuration Space has been removed starting at Base 2.0. Therefore, the test no longer supports Trusted Configuration Space.

2.1.2.18 RCRB Detection and Testing

For a function within a Root Complex, a Root Complex Register Block (RCRB) can be associated with the function. An RCRB can also be associated with another RCRB. The RCRB can be used to control multiple functions within the Root Complex. An RCRB contains registers that are similar to the registers normally defined to occupy extended configuration space in a PCI Express function, except for the following differences:

- ❑ An RCRB always resides in Memory Space.
- ❑ An RCRB's base address is given by a Link Address in a valid Link Entry of the appropriate Element Type.
- ❑ Offset 000h in the RCRB contains either an extended capability header, or else it contains 000x FFFFh (where x means bits 19-16 are don't cares), indicating the RCRB is empty.
- ❑ Only a sub-set of extended capabilities are allowed to be implemented in an RCRB.

To detect an RCRB, the following steps are done by test software:

1. For the function under test, check the Device/Port Type field (PCI Express Capabilities register) and see that it is either a Root Port or a Root Complex Integrated Endpoint. Otherwise there is no RCRB associated with it.
2. Examine the Extended Capability ID field for each of the detected Extended Capabilities. Determine how many instances of the Extended Capability ID of 0005h (Root Complex Link

Declaration Extended Capability) are found. If only one is found, then an RCRB may be present. Otherwise there is no RCRB present.

3. Check the Number of Link Entries (Element Self Description register) in the Root Complex Link Declaration Extended Capability and record the number (this indicates how many Link Entries must be scanned).
4. Check each Link Entry in the Root Complex Link Declaration Extended Capability (up to the Number of Link Entries value). Each Link Entry is checked for all the following:
 - a. Either Link Valid (Link Description register) returns 1 or Associate RCRB Header (Link Description register) returns 1 (both may return 1 as well).
 - b. Link Type (Link Description register) indicates Memory Space (returns 0).

If a Link Entry contains all of these, it is one that identifies the base address of a potential RCRB. The Link Address registers (bits 63-0) is recorded as a potential RCRB base address. Once every one of the Link Entries are checked, every individual entry in the list of potential RCRB base addresses is used to perform step 5.

5. Read the memory space address location given by the potential RCRB base address and if it returns 000x FFFFh (where x means bits 19-16 are don't cares) then this RCRB is empty. Otherwise, this is the base address of an RCRB. Repeat this step until each potential RCRB base address has been checked.
6. Repeat steps 2-5, using each discovered RCRB, to see if the RCRB is associated with another RCRB. Since association is bi-directional, each individual association will have a reciprocal association in the target, therefore each individual associated RCRB should only be counted once during the association parsing process. However, all RCRB associations must be parsed in order to determine the total number of RCRBs associated with the function under test.
7. Any function that contains a link to a valid RCRB should be tested more than once, for any test that checks PCI Express Extended Configuration Space capabilities. Such a function should be first tested using only its Extended Configuration space area, and then the test case should be re-run using every RCRB (memory space) area associated with the function, or associated with other RCRBs that are themselves associated with the function.
8. Any function that is associated with a valid RCRB is only tested once, for any test that checks PCI Configuration Space capabilities. Such a function should be tested once using only its PCI Configuration space area.

2.1.2.19 Determination of Function, Base Function, Physical Function, or Virtual Function

An Endpoint function may be defined as a Function, a Base Function (BF), a Physical Function (PF), or a Virtual Function (VF). The characteristics of each function type are as follows:

- ☐ A Function returns a Vendor ID that is not FFFFh and not 0001h (CRS Software Visibility Notification), and does not contain a SR-IOV Extended Capability.
- ☐ A BF returns a Vendor ID that is not FFFFh and not 0001h (CRS Software Visibility Notification), and contains a MR-IOV Extended Capability.
- ☐ A PF returns a Vendor ID that is not FFFFh and not 0001h (CRS Software Visibility Notification), and contains a SR-IOV Extended Capability.

- ❑ A VF returns a Vendor ID that is FFFFh (at a Bus/Device/Function address where a VF is indicated to be at).

In order for the test case to determine what type of Endpoint function the function under test is, the following sequence is followed:

1. Examine the Capability ID field for each of the function's Capabilities. Determine how many instances of the Capability ID of 10h (PCI Express Capability) are found. If none are found, this is not a Function, BF, PF, or VF so return this and stop further evaluation.
2. Having found the PCI Express Capability in this function under test, read a DWORD at offset 02h (PCI Express Capabilities register) in the PCI Express Capability and check the Device/Port Type to see that it is a PCI Express Endpoint, a Legacy Endpoint or a Root Complex Integrated Endpoint. If it is not one of these, then this is not a Function, BF, PF, or VF so return this and stop further evaluation.
3. Read back a WORD at location 00h (Vendor ID register) in Configuration Space, and if FFFFh is found this is a VF so return this and stop further evaluation.
4. Examine the Extended Capability ID field for each of the detected Extended Capabilities. Determine how many instances of the Extended Capability ID of 0011h (MR-IOV Extended Capability) are found. If none are found this function is not a BF so continue with further evaluation.
5. If a MR-IOV Extended Capability was found this function is a BF so return this and stop further evaluation.
6. Examine the Extended Capability ID field for each of the detected Extended Capabilities. Determine how many instances of the Extended Capability ID of 0010h (SR-IOV Extended Capability) are found. If none are found this function is a Function so return this and stop further evaluation.
7. If a SR-IOV Extended Capability was found this function is a PF so return this and stop further evaluation.

2.1.2.20 ASPM Testing Tables

For tests that enable ASPM, the different testing levels use different interpretations of the Active State Power Management (ASPM) Support field in the Link Capabilities register. Prior to the ASPM Optionality ECN to the Base 2.1 Specification, RxL0s support was mandatory and this field only had three settings: No ASPM; TxL0s Supported; L1+TxL0s Supported. After this ECN, the field had four settings: No ASPM; TxL0s+RxL0s Supported; L1 Supported; L1+TxL0s+RxL0s Supported. This difference in interpretations means that the different ASPM test combinations are defined, depending on the implementation of the port on each side of the link. The test combinations are listed in the following tables and are referenced by the applicable test descriptions.

2.1.2.20.1 Table 1 – Neither Port Supports ASPM Optionality

This table lists the combinations of ASPM settings that will be tested when the ports on both sides of the link do not support ASPM Optionality (ASPM Optionality Compliance field in Link Capabilities register reports 0). The table uses two values for the Active State Power Management [ASPM] Support field in the Link Capabilities register: [UASPM] which is the value returned by the

upstream port of the link and [DASPM] which is the value returned by the downstream port of the link.

Upstream Port of Link	Downstream Port of Link
Disabled	L0s only (If [DASPM] = 01b or 11b)
L0s only (If [UASPM] = 01b or 11b)	Disabled
L0s only (If [UASPM] = 01b or 11b)	L0s only (If [DASPM] = 01b or 11b)
L1 only (If [UASPM] = 11b)	Disabled
L1 only (If [UASPM] = 11b)	L0s only (If [DASPM] = 01b or 11b)
L1 only (If [UASPM] = 11b)	L1 only (If [DASPM] = 11b)
L1 only (If [UASPM] = 11b)	L0s and L1 (If [DASPM] = 11b)
L0s and L1 (If [UASPM] = 11b)	Disabled
L0s and L1 (If [UASPM] = 11b)	L0s only (If [DASPM] = 01b or 11b)
L0s and L1 (If [UASPM] = 11b)	L1 only (If [DASPM] = 11b)
L0s and L1 (If [UASPM] = 11b)	L0s and L1 (If [DASPM] = 11b)

Table 1: Neither Port Supports ASPM Optionality

2.1.2.20.2 Table 2 – Only Upstream Port Supports ASPM Optionality

This table lists the combinations of ASPM settings that will be tested when the upstream port on the link supports ASPM Optionality (ASPM Optionality Compliance field in Link Capabilities register reports 1), but the downstream port of the link does not. The table uses two values for the Active State Power Management [ASPM] Support field in the Link Capabilities register: [UASPM] which is the value returned by the upstream port of the link and [DASPM] which is the value returned by the downstream port of the link.

Upstream Port of Link	Downstream Port of Link
Disabled	L0s only (If [DASPM] = 01b or 11b)
L0s only (If ([UASPM] = 01b or 11b) and ([DASPM] = 01b or 11b))	Disabled
L0s only (If ([UASPM] = 01b or 11b) and ([DASPM] = 01b or 11b))	L0s only (If [DASPM] = 01b or 11b)
L1 only (If [UASPM] = 10b or 11b)	Disabled
L1 only (If [UASPM] = 10b or 11b)	L0s only (If [DASPM] = 01b or 11b)
L1 only (If [UASPM] = 10b or 11b)	L1 only (If [DASPM] = 11b)
L1 only (If [UASPM] = 10b or 11b)	L0s and L1 (If [DASPM] = 11b)
L0s and L1 (If ([UASPM] = 11b) and ([DASPM] = 01b or 11b))	Disabled
L0s and L1 (If ([UASPM] = 11b) and ([DASPM] = 01b or 11b))	L0s only (If [DASPM] = 01b or 11b)
L0s and L1 (If ([UASPM] = 11b) and ([DASPM] = 01b or 11b))	L1 only (If [DASPM] = 11b)
L0s and L1 (If ([UASPM] = 11b) and ([DASPM] = 01b or 11b))	L0s and L1 (If [DASPM] = 11b)

Table 2: Only Upstream Port Supports ASPM Optionality

2.1.2.20.3 Table 3 – Only Downstream Port Supports ASPM Optionality

This table lists the combinations of ASPM settings that will be tested when the downstream port on the link supports ASPM Optionality (ASPM Optionality Compliance field in Link Capabilities register reports 1), but the upstream port of the link does not. The table uses two values for the Active State Power Management [ASPM] Support field in the Link Capabilities register: [UASPM] which is the value returned by the upstream port of the link and [DASPM] which is the value returned by the downstream port of the link.

Upstream Port of Link	Downstream Port of link
Disabled	L0s only (If ([DASPM] = 01b or 11b) and ([UASPM] = 01b or 11b))
L0s only (If [UASPM] = 01b or 11b)	Disabled
L0s only (If [UASPM] = 01b or 11b)	L0s only (If ([DASPM] = 01b or 11b) and ([UASPM] = 01b or 11b))
L1 only (If [UASPM] = 11b)	Disabled
L1 only (If [UASPM] = 11b)	L0s only (If ([DASPM] = 01b or 11b) and ([UASPM] = 01b or 11b))
L1 only (If [UASPM] = 11b)	L1 only (If [DASPM] = 10b or 11b)
L1 only (If [UASPM] = 11b)	L0s and L1 (If ([DASPM] = 11b) and ([UASPM] = 01b or 11b))
L0s and L1 (If [UASPM] = 11b)	Disabled
L0s and L1 (If [UASPM] = 11b)	L0s only (If ([DASPM] = 01b or 11b) and ([UASPM] = 01b or 11b))
L0s and L1 (If [UASPM] = 11b)	L1 only (If [DASPM] = 10b or 11b)
L0s and L1 (If [UASPM] = 11b)	L0s and L1 (If ([DASPM] = 11b) and ([UASPM] = 01b or 11b))

Table 3: Only Downstream Port Supports ASPM Optionality

2.1.2.20.4 Table 4 – Both Ports Support ASPM Optionality

This table lists the combinations of ASPM settings that will be tested when both of the ports on the link support ASPM Optionality (ASPM Optionality Compliance field in Link Capabilities register reports 1). The table uses two values for the Active State Power Management [ASPM] Support field in the Link Capabilities register: [UASPM] which is the value returned by the upstream port of the link and [DASPM] which is the value returned by the downstream port of the link.

Upstream Port of Link	Downstream Port of Link
Disabled	L0s only (If ([DASPM] = 01b or 11b) and ([UASPM] = 01b or 11b))
L0s only (If ([UASPM] = 01b or 11b) and ([DASPM] = 01b or 11b))	Disabled
L0s only (If ([UASPM] = 01b or 11b) and ([DASPM] = 01b or 11b))	L0s only (If ([DASPM] = 01b or 11b) and ([UASPM] = 01b or 11b))
L1 only (If [UASPM] = 10b or 11b)	Disabled
L1 only (If [UASPM] = 10b or 11b)	L0s only (If ([DASPM] = 01b or 11b) and ([UASPM] = 01b or 11b))
L1 only (If [UASPM] = 10b or 11b)	L1 only (If [DASPM] = 10b or 11b)
L1 only (If [UASPM] = 10b or 11b)	L0s and L1 (If ([DASPM] = 11b) and ([UASPM] = 01b or 11b))
L0s and L1 (If ([UASPM] = 11b) and ([DASPM] = 01b or 11b))	Disabled
L0s and L1 (If ([UASPM] = 11b) and ([DASPM] = 01b or 11b))	L0s only (If ([DASPM] = 01b or 11b) and ([UASPM] = 01b or 11b))
L0s and L1 (If ([UASPM] = 11b) and ([DASPM] = 01b or 11b))	L1 only (If [DASPM] = 10b or 11b)
L0s and L1 (If ([UASPM] = 11b) and ([DASPM] = 01b or 11b))	L0s and L1 (If ([DASPM] = 11b) and ([UASPM] = 01b or 11b))

Table 4: Both Ports Support ASPM Optionality

2.2. Configuration Register Tests (All Devices)

2.2.1. TD_1_2 PCI Express Capability Structure

The test verifies that if the function under test reports a PCI Express Capability structure, it implements the required registers for the device/port type and capability version as defined in the relevant specifications. (The individual register field contents are not checked, they will be checked by other tests.)

Relevant Specifications

- ☐ *PCI Express Base Specification*
- ☐ *ECN PCI Express Capability Structure Expansion (to Base 1.1)*

(See Section 1.3 for additional specification revision details.)

Applicable Device/Port Types

This test is run on all device/port types.

Starting Configuration

This test is run with functions starting in the D0-Uninitialized state following the standard initialization procedures in Section 2.1.1. Testing in additional Dx states may be optionally supported. ASPM states of the upstream and downstream ports are disabled for this test.

Overview of Test Steps

Test software performs the following steps:

1. Configure the function under test following the procedure described in Section 2.1.1.
2. Examine the Capability ID field for each of the function's Capabilities. Determine how many instances of the Capability ID of 10h (PCI Express Capability) are found. If more than one is found, the test terminates with a failure.
3. If the Capability ID of 10h is not found for a capability, the test terminates with a failure.
4. If a Capability ID of 10h is found for a capability, read the DWORD at offset 00h, and use the contents of the PCI Express Capabilities register (Capability Version field, Device/Port Type field, and Slot Implemented field) to determine the total size of the PCI Express Capability structure.
5. Attempt to read the indicated size from each of the following additional offsets (test software records if any of these read attempts fail to complete, but does not check the value returned):
 - a. If the Capability Version field is 1h or less:
 - i. If the Device/Port Type field is 0h (Endpoint), or 01h (Legacy Endpoint), or 05h (Switch Upstream Port), or 07h (PCI Express to PCI/PCI-X Bridge): 04h (DWORD); 08h (WORD); 0Ah (WORD); 0Ch (DWORD); 10h (WORD).
 - ii. If the Device/Port Type field is 4h (Root Port) and Slot Implemented is 0: 04h (DWORD); 08h (WORD); 0Ah (WORD); 0Ch (DWORD); 10h (WORD); 1Ch (WORD); 1Eh (WORD); 20h (DWORD).
 - iii. If the Device/Port Type field is 4h (Root Port) and Slot Implemented is 1: 04h (DWORD); 08h (WORD); 0Ah (WORD); 0Ch (DWORD); 10h (WORD); 14h (DWORD); 18h (WORD); 1Ah (WORD); 1Ch (WORD); 1Eh (WORD); 20h (DWORD).
 - iv. If the Device/Port Type field is 6h (Switch Downstream Port) and Slot Implemented is 0, or 8h (PCI/PCI-X to PCI Express Bridge) and Slot Implemented is 0: 04h (DWORD); 08h (WORD); 0Ah (WORD); 0Ch (DWORD); 10h (WORD).
 - v. If the Device/Port Type field is 6h (Switch Downstream Port) and Slot Implemented is 1, or 8h (PCI/PCI-X to PCI Express Bridge) and Slot Implemented is 1: 04h (DWORD); 08h (WORD); 0Ah (WORD); 0Ch (DWORD); 10h (WORD); 14h (DWORD); 18h (WORD); 1Ah (WORD).
 - vi. If the Device/Port Type field is 9h (Root Complex Integrated Endpoint): 04h (DWORD); 08h (WORD); 0Ah (WORD).
 - vii. If the Device/Port Type field is Ah (Root Complex Event Collector): 04h (DWORD); 08h (WORD); 0Ah (WORD); 1Ch (WORD); 1Eh (WORD); 20h (DWORD).

- b. If the Capability Version field is 2h or greater: 00h (DWORD); 04h (DWORD); 08h (WORD); 0Ah (WORD); 0Ch (DWORD); 10h (WORD); 12h (WORD); 14h (DWORD); 18h (WORD); 1Ah (WORD); 1Ch (WORD); 1Eh (WORD); 20h (DWORD); 24h (DWORD); 28h (WORD); 2Ah (WORD); 2Ch (DWORD); 30h (WORD); 32h (WORD); 34h (DWORD); 38h (WORD); 3Ah (WORD).
- 6. The Next Capability Pointer field must be 00h or greater than 3Fh and the lower 2 bits of this field must be 00b.
- 7. For functions under test that have a link, the test is run at each of the following link speeds:
 - a. 2.5 GT/s (for devices that are capable). See Section 2.1.2.12 for details.
 - b. 5.0 GT/s (for devices that are capable). See Section 2.1.2.13 for details.
 - c. 8.0 GT/s (for devices that are capable). See Section 2.1.2.14 for details.

The test *fails* if:

- ☐ A PCI Express capability structure is not present.
- ☐ More than one PCI Express Capability is present.
- ☐ A non-zero Next Capability Pointer is less than or equal to 3Fh, or the lower 2 bits are non-zero.
- ☐ A valid register access within the PCI Express Capability structure fails to complete.

2.2.2. TD_1_3 PCI Express Capabilities Register

The test verifies that the function under test implements the PCI Express Capabilities register as defined in the relevant specifications.

Relevant Specifications

- ☐ *PCI Express Base Specification*
- ☐ *ECN Trusted Configuration Space (to Base 1.1)*
- ☐ *PCI Local Bus Specification, Revision 3.0*

(See Section 1.3 for additional specification revision details.)

Applicable Device/Port Types

This test is run on all device/port types.

Starting Configuration

This test is run with functions starting in the D0-Uninitialized state following the standard initialization procedures in Section 2.1.1. Testing in additional Dx states may be optionally supported. ASPM states of the upstream and downstream ports are disabled for this test.

Overview of Test Steps

Test software performs the following steps:

1. Configure the function under test following the procedure described in Section 2.1.1.
2. Examine the Capability ID field for each of the function's Capabilities. Determine how many instances of the Capability ID of 10h (PCI Express Capability) are found. If more than one is found, the test terminates with a failure.
3. If the Capability ID of 10h is not found for a capability, the test terminates with a failure.

4. If a Capability ID of 10h is found for a capability, the following tests are performed:
5. Read a WORD located at offset 02h in the PCI Express Capability Structure.
6. Perform each of the following checks on the fields of the WORD read:
 - a. For Base 1.x testing: the Capability Version field must be 1h or 2h.
 - b. For Base 2.x or later testing: the Capability Version field must be 2h.
 - c. The Device/Port Type field must be 0000b, 0001b, 0100b, 0101b, 0110b, 0111b, 1000b, 1001b, or 1010b. All other encodings are reserved and treated as a failure.
7. Read a byte at location 0Eh (Header Type register) and check bits 6-0 to see that they are 000 0000b (Type 0) or 000 0001b (Type 1). All other encodings are reserved and treated as a failure. If the device implements a valid Header Type the following checks are performed:
 - a. For Type 0 Configuration Space header, the Device/Port Type field must be 0000b (Endpoint), 0001b (Legacy Endpoint), 1001b (Root Complex Integrated Endpoint), or 1010b (Root Complex Event Collector).
 - b. For Type 1 Configuration Space header, the Device/Port Type field must be 0100b (Root Port), 0101b (Switch Upstream Port), 0110b (Switch Downstream Port), 0111b (PCI Express to PCI/PCI-X Bridge), or 1000b (PCI/PCI-X to PCI Express Bridge).
8. If the Slot Implemented field returns 1, the Device/Port Type field must be 0100b (Root Port), 0110b (Switch Downstream Port), or 1000b (PCI/PCI-X to PCI Express Bridge).
9. The following register field characteristic checks are performed:

PCI Express Capability Header (Offset 00h) — WORD

- | | |
|----------------------------|----|
| a. Capability ID | RO |
| b. Next Capability Pointer | RO |

PCI Express Capabilities Register (Offset 02h) — DWORD

- | | |
|--|---------|
| a. Capability Version | RO |
| b. Device/Port Type | RO |
| c. Slot Implemented
(for non-FLR testing) | HwInit |
| (for FLR testing) | RO |
| d. Interrupt Message Number | RO |
| e. Undefined_14 | RO |
| Note: This was once TCS Routing Supported. | |
| f. RsvdP_15 | RO-Zero |

10. For functions under test that have a link, the test is run at each of the following link speeds:
 - a. 2.5 GT/s (for devices that are capable). See Section 2.1.2.12 for details.
 - b. 5.0 GT/s (for devices that are capable). See Section 2.1.2.13 for details.
 - c. 8.0 GT/s (for devices that are capable). See Section 2.1.2.14 for details.

The test *fails* if:

- ☐ A PCI Express capability structure is not present.
- ☐ More than one PCI Express Capability is present.
- ☐ The Capability Version field is not 1h or 2h (for a Base 1.x compliant test).
- ☐ The Capability Version field is not 2h (for a Base 2.x or later compliant test).

- ☐ The Device/Port Type field is not one of the defined values.
- ☐ The Header Type field is not a Type 0 or Type 1 Configuration Space Header Type.
- ☐ The PCI Configuration Space Header Type does not match the requirements of the Device/Port Type.
- ☐ The Slot Implemented field is 1 and the Device/Port Type field is not Root Port, Switch Downstream Port, or PCI/PCI-X to PCI Express Bridge.
- ☐ Any of the register field characteristic tests fail.

2.2.3. TD_1_4 Device Capabilities, Device Control, and Device Status Registers

The test verifies that the function under test implements the PCI Express Device Capabilities, Device Control, and Device Status registers as defined in the relevant specifications.

Relevant Specifications

- ☐ *PCI Express Base Specification*
- ☐ *ECN Function Level Reset (to Base 1.1)*
- ☐ *ECN Alternate Routing-ID Interpretation (to Base 2.0 and Base 1.1)*
- ☐ *ECN Extended Tag Enable Default (to Base 2.0)*
- ☐ *Single Root I/O Virtualization and Sharing Specification*
- ☐ *Multi-Root I/O Virtualization and Sharing Specification Revision 1.0*

(See Section 1.3 for additional specification revision details.)

Applicable Device/Port Types

This test is run on all device/port types.

Starting Configuration

This test is run with functions starting in the D0-Uninitialized state following the standard initialization procedures in Section 2.1.1. Testing in additional Dx states may be optionally supported. ASPM states of the upstream and downstream ports are disabled for this test.

Overview of Test Steps

Test software performs the following steps:

1. Configure the function under test following the procedure described in Section 2.1.1.
2. Examine the Capability ID field for each of the function's Capabilities. Determine how many instances of the Capability ID of 10h (PCI Express Capability) are found. If more than one is found, the test terminates with a failure.
3. If the Capability ID of 10h is not found for a capability, the test terminates with a failure.
4. If a Capability ID of 10h is found for a capability, the following tests are performed:
5. Read a DWORD located at offset 04h (Device Capabilities register) in the PCI Express Capability structure.

6. Perform each of the following checks on the fields of the DWORD read:
7. The Max_Payload_Size Supported field must not return 110b or 111b (Reserved), if it does the test fails. Test software writes the value read from the Max_Payload_Size Supported field into the Max_Payload_Size field. Test software then reads back the Max_Payload_Size field and ensures that the written value is returned. This test is repeated for each value (including 0) less than the value read from the Max_Payload_Size Supported field.
8. The following checks are performed on the Phantom Functions Supported field:
 - a. Returned value is 01b:
 - i. The function must not be part of a multi-function device using functions 4, 5, 6, or 7.
 - ii. The function must not be a function of an ARI device.
 - iii. The function must not be a VF.
 - b. Returned value is 10b:
 - i. The function must not be part of a multi-function device using functions 2, 3, 4, 5, 6, or 7.
 - ii. The function must not be a function of an ARI device.
 - iii. The function must not be a VF.
 - c. Returned value is 11b:
 - i. The function must not be part of a multi-function device.
 - ii. The function must not be a function of an ARI device.
 - iii. The function must not be a VF.
9. If the Function Level Reset Capability field returns 1, the Device/Port Type field (PCI Express Capabilities register) must be 0000b (Endpoint), 0001b (Legacy Endpoint), or 1001b (Root Complex Integrated Endpoint).
10. The following register field characteristic checks are performed:

Device Capabilities Register (Offset 04h) — DWORD

a. Max_Payload_Size Supported	RO
b. Phantom Functions Supported (for VFs and ARI functions) (otherwise)	RO-Zero RO
c. Extended Tag Field Supported	RO
d. Endpoint L0s Acceptable Latency (for Endpoints) (for non-Endpoints)	RO RO-Zero
e. Endpoint L1 Acceptable Latency (for Endpoints) (for non-Endpoints)	RO RO-Zero
f. Undefined_12 Note: This was once Attention Button Present.	RO
g. Undefined_13 Note: This was once Attention Indicator Present.	RO
h. Undefined_14 Note: This was once Power Indicator Present.	RO
i. Role-Based Error Reporting	RO-Ones
j. RsvdP_17-16	RO-Zero
k. Captured Slot Power Limit Value	

(for Endpoints excluding VFs)	RO
(for Switch Upstream Ports)	RO
(for PCI Express to PCI/PCI-X Bridges)	RO
l. Captured Slot Power Limit Scale	
(for Endpoints excluding VFs)	RO
(for Switch Upstream Ports)	RO
(for PCI Express to PCI/PCI-X Bridges)	RO
m. Function Level Reset Capability	
(for BFs, PFs and VFs)	RO-Ones
(for Endpoints except BFs, PFs and VFs)	RO
(for non-Endpoints)	RO-Zero
n. RsvdP_31-29	RO-Zero

Device Control Register (Offset 08h) — WORD

a. Correctable Error Reporting Enable	
(for RC Integrated Endpoints)	RW or
	RO-Zero
(for VFs)	RO-Zero
(for all others)	RW
b. Non-Fatal Error Reporting Enable	
(for RC Integrated Endpoints)	RW or
	RO-Zero
(for VFs)	RO-Zero
(for all others)	RW
c. Fatal Error Reporting Enable	
(for RC Integrated Endpoints)	RW or
	RO-Zero
(for VFs)	RO-Zero
(for all others)	RW
d. Unsupported Request Reporting Enable	
(for RC Integrated Endpoints)	RW or
	RO-Zero
(for VFs)	RO-Zero
(for all others)	RW
e. Enable Relaxed Ordering	
(for VFs)	RO-Zero
(otherwise)	RW or
	RO-Zero
f. Max_Payload_Size	
(for VFs)	RO-Zero
(otherwise if Max_Payload_Size Supported is 000b)	RW or
	RO-Zero
(otherwise if Max_Payload_Size Supported is non-zero)	RW
g. Extended Tag Field Enable	
(if Extended Tag Field Supported is 1)	RW
(if Extended Tag Field Supported is 0)	RO-Zero
h. Phantom Functions Enable	
(if Phantom Functions Supported is 1)	RW

- | | |
|---|-------------------------------------|
| (if Phantom Functions Supported is 0) | RO-Zero |
| i. Auxiliary (AUX) Power PM Enable
(for VFs)
(otherwise, if Aux Current is non-zero in PM Capability)
(otherwise if Aux Current is 000b in PM Capability) | RO-Zero
RWS
RWS or
RO-Zero |
| j. Enable No Snoop
(for VFs)
(otherwise) | RO-Zero
RW or
RO-Zero |
| k. Max_Read_Request_Size
(for VFs)
(otherwise) | RO-Zero
RW or
RO-Zero |
| l. Bridge Configuration Retry Enable
/Initiate Function Level Reset
(for PCI Express to PCI/PCI-X Bridge)
(for Endpoints with Function Level Reset Capability as 1)
Note: For Endpoints, writing the Initiate Function Level Reset field causes the function to
reset to its default state if supported (the Function Level Reset Capability field returns 1).
Test software must wait 100 ms after writing this field to 1 (if the Function Level Reset
Capability field returns 1), before reading any register in this function. The Initiate Function
Level Reset field always returns 0 when read. | RW
RO-Zero |
| m. RsvdP_15
(for Endpoints with Function Level Reset Capability as 0)
(for non-Endpoints and non-PCI Express to PCI/PCI-X Bridges) | RO-Zero
RO-Zero |

Device Status Register (Offset 0Ah) — WORD

- | | |
|---|---------------|
| a. Correctable Error Detected | RW1C |
| b. Non-Fatal Error Detected | RW1C |
| c. Fatal Error Detected | RW1C |
| d. Unsupported Request Detected | RW1C |
| e. AUX Power Detected
(for VFs)
(otherwise) | RO-Zero
RO |
| f. Transactions Pending | RO |
| g. RsvdZ_15-6 | RO-Zero |

11. The following default value checks are performed:

Device Control Register Default Value (Offset 08h) — WORD

- | | |
|--|------|
| a. Correctable Error Reporting Enable | 0 |
| b. Non-Fatal Error Reporting Enable | 0 |
| c. Fatal Error Reporting Enable | 0 |
| d. Unsupported Request Reporting Enable | 0 |
| e. Enable Relaxed Ordering
(for non-VFs with Enable Relaxed Ordering as RW) | 1 |
| f. Max_Payload_Size
(for functions with a link) | 000b |

- | | |
|---|-----------------------------|
| g. Extended Tag Field Enable
(if Extended Tag Field Supported is 1) | 1 or 0
(Based on design) |
| h. Phantom Functions Enable | 0 |
| i. Enable No Snoop
(for non-VFs with Enable No Snoop as RW) | 1 |
| j. Max_Read_Request_Size
(for non-VFs with Max_Read_Request_Size as RW) | 010b |
| k. Bridge Configuration Retry Enable/Initiate Function Level Reset
(for PCI Express to PCI/PCI-X Bridge) | 0 |

Device Status Register Default Value (Offset 0Ah) — WORD

- | | |
|---------------------------------|---|
| a. Correctable Error Detected | 0 |
| b. Non-Fatal Error Detected | 0 |
| c. Fatal Error Detected | 0 |
| d. Unsupported Request Detected | 0 |
| e. Transactions Pending | 0 |
12. For functions under test that have a link, the test is run at each of the following link speeds:
- a. 2.5 GT/s (for devices that are capable). See Section 2.1.2.12 for details.
 - b. 5.0 GT/s (for devices that are capable). See Section 2.1.2.13 for details.
 - c. 8.0 GT/s (for devices that are capable). See Section 2.1.2.14 for details.

The test *fails* if:

- ☐ A PCI Express capability structure is not present.
- ☐ More than one PCI Express Capability is present.
- ☐ A supported value written to the Max_Payload_Size control field is not read back.
- ☐ A reserved value is read from the Max_Payload_Size Supported field.
- ☐ The function under test is part of an ARI device, a VF function, or is part of a multi-function device that makes the reported non-zero value in the Phantom Functions Supported field invalid.
- ☐ Any of the register field characteristic tests fail.
- ☐ Any of the default value tests fail.

2.2.4. TD_1_40 Device Capabilities 2, Device Control 2, and Device Status 2 Registers (PCIe Cap Ver = 2)

The test verifies that the function under test implements the PCI Express Device Capabilities 2, Device Control 2 and Device Status 2 registers as defined in the relevant specifications.

Relevant Specifications

- ☐ *PCI Express Base Specification*
- ☐ *ECN PCI Express Capability Structure Expansion (to Base 1.1)*
- ☐ *ECN Completion Timeout Control Capability (to Base 1.1)*
- ☐ *ECN Alternate Routing-ID Interpretation (to Base 2.0 and Base 1.1)*

- ❑ *ECN Atomic Operations (to Base 2.0)*
- ❑ *ECN ID-Based Ordering (to Base 2.0)*
- ❑ *ECN Latency Tolerance Reporting (to Base 2.0)*
- ❑ *ECN TLP Prefix (to Base 2.0)*
- ❑ *ECN TLP Processing Hints (to Base 2.0)*
- ❑ *ECN Optimized Buffer Flush/Fill (to Base 2.0 and Base 2.1)*
- ❑ *Single Root I/O Virtualization and Sharing Specification*

(See Section 1.3 for additional specification revision details.)

Applicable Device/Port Types

This test is run on all device/port types.

Starting Configuration

This test is run with functions starting in the D0-Uninitialized state following the standard initialization procedures in Section 2.1.1. Testing in additional Dx states may be optionally supported. ASPM states of the upstream and downstream ports are disabled for this test.

Overview of Test Steps

Test software performs the following steps:

1. Configure the function under test following the procedure described in Section 2.1.1.
2. Examine the Capability ID field for each of the function's Capabilities. Determine how many instances of the Capability ID of 10h (PCI Express Capability) are found. If more than one is found, the test terminates with a failure.
3. If the Capability ID of 10h is not found for a capability, the test terminates with a failure.
4. If a Capability ID of 10h is found for a capability, read a WORD located at offset 02h in the PCI Express Capability Structure and if the Capability Version field is equal or less than 1h, the test terminates. (For Base 2.x or later testing, this is a failure. For Base 1.x testing this is not a failure.)
5. If a Capability ID of 10h is found for a capability and if the Capability Version field is equal or greater than 2h, the following tests are performed:
6. Read a DWORD located at offset 24h (Device Capabilities 2 register) in the PCI Express Capability Structure.
7. Perform each of the following checks on the fields of the DWORD read:
 - a. The Completion Timeout Ranges Supported field must be a valid value (0000b, 0001b, 0010b, 0011b, 0110b, 0111b, 1110b, or 1111b).
 - b. For Base 2.x or later testing: if the device type is a Root Port, if the 32-bit AtomicOp Completer Supported field returns 1, then the 64-bit AtomicOp Completer Supported field must return 1.
 - c. For Base 2.x or later testing: if the device type is a Root Port, if the 128-bit CAS Completer Supported field returns 1, then both the 32-bit AtomicOp Completer Supported field and the 64-bit AtomicOp Completer Supported field must return 1.

8. The following register field characteristic checks are performed:

Device Capabilities 2 Register (Offset 24h) — DWORD

- | | |
|--|-----------------------------|
| a. Completion Timeout Ranges Supported
(for Switch Ports, RC Event Collectors,
or PCI/PCI-X to PCI Express Bridges)
(otherwise, for non-FLR testing)
(otherwise, for FLR testing) | RO-Zero
HwInit
RO |
| b. Completion Timeout Disable Supported
(for Switch Ports, RC Event Collectors,
or PCI/PCI-X to PCI Express Bridges)
(otherwise) | RO-Zero
RO |
| c. ARI Forwarding Supported
(for Switch Downstream Ports or Root Ports)
(otherwise) | RO
RO-Zero |
| d. AtomicOp Routing Supported
(For Base 2.x or later testing)
(for Switch Upstream Ports, Switch Downstream Ports, or Root Ports)
(otherwise) | RO
RO-Zero |
| e. 32-bit AtomicOp Completer Supported
(For Base 2.x or later testing)
(for Endpoints or Root Ports)
(otherwise) | RO
RO-Zero |
| f. 64-bit AtomicOp Completer Supported
(For Base 2.x or later testing)
(for Endpoints or Root Ports)
(otherwise) | RO
RO-Zero |
| g. 128-bit CAS Completer Supported
(For Base 2.x or later testing)
(for Endpoints or Root Ports)
(otherwise) | RO
RO-Zero |
| h. No RO-enabled PR-PR Passing
(For Base 2.x or later testing)
(for Switch Upstream Ports, Switch Downstream Ports,
or Root Ports, for non-FLR testing)
(for Switch Upstream Ports, Switch Downstream Ports,
or Root Ports, for FLR testing)
(otherwise) | HwInit

RO
RO-Zero |
| i. LTR Mechanism Supported
(For Base 2.x or later testing)
(for PCI Express to PCI/PCI-X Bridges
or PCI/PCI-X to PCI Express Bridges)
(otherwise) | RO-Zero
RO |
| j. TPH Completer Supported
(For Base 2.x or later testing)
(for Endpoints or Root Ports)
(otherwise) | RO
RO-Zero |
| k. RsvdP_17-14 | |

	(For Base 2.x or later testing)	RO-Zero
l.	OBFF Supported (For Base 2.x or later testing) (for Root Ports, Switch Ports, Endpoints, for non-FLR testing) (for Root Ports, Switch Ports, Endpoints, for FLR testing) (otherwise)	HwInit RO RO-Zero
m.	Extended Fmt Field Supported (For Base 2.x or later testing)	RO
n.	End-End TLP Prefix Supported (For Base 2.x or later testing) (for Root Ports, for non-FLR testing) (otherwise)	HwInit RO
o.	Max End-End TLP Prefixes (For Base 2.x or later testing) (for Switch Upstream Ports or Switch Downstream Ports) (for non-Switches with End-End TLP Prefix Supported as 0) (for Root Ports with End-End TLP Prefix Supported as 1, for non-FLR testing) (for Root Ports with End-End TLP Prefix Supported as 1, for FLR testing) (for non-Switches and non-Root Ports, with End-End TLP Prefix Supported as 1)	RO-Zero RO-Zero HwInit RO RO
p.	RsvdP_31-24 (For Base 2.x or later testing)	RO-Zero
q.	RsvdP_31-6 (For Base 1.x testing)	RO-Zero
Device Control 2 Register (Offset 28h) — WORD		
a.	Completion Timeout Value (for VFs, Switch Ports, RC Event Collectors, or PCI/PCI-X to PCI Express Bridges) (for Root Ports, Endpoints (not VFs), or PCI Express to PCI/PCI-X Bridges)	RO-Zero RO-Zero or RW
b.	Completion Timeout Disable (for VFs, Switch Ports, RC Event Collectors, or PCI/PCI-X to PCI Express Bridges) (for Root Ports, Endpoints (not VFs), or PCI Express to PCI/PCI-X Bridges)	RO-Zero RO-Zero or RW
c.	ARI Forwarding Enable (for Switch Downstream Ports with ARI Forwarding Supported as 1) (for Root Ports with ARI Forwarding Supported as 1) (otherwise)	RW RW RO-Zero
d.	AtomicOp Requester Enable (For Base 2.x or later testing) (for Endpoints (not VFs) or Root Ports)	RW or RO-Zero

(otherwise)	RO-Zero
e. AtomicOp Egress Blocking (For Base 2.x or later testing) (for Switch Ports with AtomicOp Routing Supported as 1) (for Root Ports with AtomicOp Routing Supported as 1) (otherwise)	RW RW RO-Zero
f. IDO Request Enable (For Base 2.x or later testing) (for Endpoints (not VFs) or Root Ports) (otherwise)	RW or RO-Zero RO-Zero
g. IDO Completion Enable (For Base 2.x or later testing) (for Endpoints (not VFs) or Root Ports) (otherwise)	RW or RO-Zero RO-Zero
h. LTR Mechanism Enable (For Base 2.x or later testing) (for PCI Express to PCI/PCI-X Bridges or PCI/PCI-X to PCI Express Bridges) (for Endpoints in single function device with LTR Mechanism Supported as 1) (for Endpoints in multi-function device that is function 0 with LTR Mechanism Supported as 1) (for Endpoints in multi-function device that is not function 0) (for Root Ports or Switch Ports with LTR Mechanism Supported as 1) (for Root Ports or Switch Ports or Endpoints, with LTR Mechanism Supported as 0)	RO-Zero RW RW RO-Zero RW RO-Zero
i. RsvdP_12-11 (For Base 2.x or later testing)	RO-Zero
j. OBFF Enable (For Base 2.x or later testing) (for Root Ports with OBFF Supported as 1) (for Switch Ports with OBFF Supported as 1) (for Endpoints in single function device with OBFF Supported as 1) (for Endpoints in multi-function device with OBFF Supported as 1 that is function 0) (for Endpoints in multi-function device with OBFF Supported as 1 that is not function 0) (otherwise)	RW RW RW RW RO-Zero RW or RO-Zero
k. End-End TLP Prefix Blocking (For Base 2.x or later testing) (for Switch Ports with End-End TLP Prefix Supported as 1) (for Root Ports with End-End TLP Prefix Supported as 1) (otherwise)	RW RW or RO-Ones RO-Zero

- l. RsvdP_15-6
(For Base 1.x testing: bits 15-6) RO-Zero

Device Status 2 Register (Offset 2Ah) — WORD

- a. RsvdZ_15-0 RO-Zero

9. The following default value checks are performed:

Device Control 2 Register Default Value (Offset 28h) — WORD

- | | |
|---|-------|
| a. Completion Timeout Value | 0000b |
| b. Completion Timeout Disable | 0 |
| c. ARI Forwarding Enable | 0 |
| d. AtomicOp Requester Enable
(For Base 2.x or later testing) | 0 |
| e. AtomicOp Egress Blocking
(For Base 2.x or later testing) | 0 |
| f. IDO Request Enable
(For Base 2.x or later testing) | 0 |
| g. IDO Completion Enable
(For Base 2.x or later testing) | 0 |
| h. LTR Mechanism Enable
(For Base 2.x or later testing) | 0 |
| i. OBFF Enable
(For Base 2.x or later testing) | 00b |
| j. End-End TLP Prefix Blocking
(For Base 2.x or later testing)
(for non-Root Ports) | 0 |

10. For functions under test that have a link, the test is run at each of the following link speeds:
- 2.5 GT/s (for devices that are capable). See Section 2.1.2.12 for details.
 - 5.0 GT/s (for devices that are capable). See Section 2.1.2.13 for details.
 - 8.0 GT/s (for devices that are capable). See Section 2.1.2.14 for details.

The test *fails* if:

- ☐ A PCI Express capability structure is not present.
- ☐ More than one PCI Express Capability is present.
- ☐ A reserved value is read from the Completion Timeout Ranges Supported field.
- ☐ A Root Port reports the 32-bit AtomicOp Completer Supported field as 1, but the 64-bit AtomicOp Completer Supported field as 0 (for Base 2.x or later testing only).
- ☐ A Root Port reports the 64-bit AtomicOp Completer Supported field as 1, but the 32-bit AtomicOp Completer Supported field as 0 (for Base 2.x or later testing only).
- ☐ A Root Port reports the 128-bit CAS Completer Supported field as 1, but both the 32-bit AtomicOp Completer Supported field as 0 and the 64-bit AtomicOp Completer Supported field as 0 (for Base 2.x or later testing only).
- ☐ Any of the register field characteristic tests fail.
- ☐ Any of the default value tests fail.

2.2.5. TD_1_5 Link Capabilities, Link Control, and Link Status Registers

The test verifies that the function under test implements the PCI Express Link Capabilities, Link Control, and Link Status registers as defined in the relevant specifications.

Relevant Specifications

- ☐ *PCI Express Base Specification*
- ☐ *ECN Link Bandwidth Notification (to Base 1.1)*
- ☐ *ECN ASPM Optionality (to Base 2.1)*
- ☐ *Single Root I/O Virtualization and Sharing Specification*

(See Section 1.3 for additional specification revision details.)

Applicable Device/Port Types

This test is run on all device/port types.

Starting Configuration

This test is run with functions starting in the D0-Uninitialized state following the standard initialization procedures in Section 2.1.1. Testing in additional Dx states may be optionally supported. ASPM states of the upstream and downstream ports are disabled for this test.

Overview of Test Steps

Test software performs the following steps:

1. Configure the function under test following the procedure described in Section 2.1.1.
2. Examine the Capability ID field for each of the function's Capabilities. Determine how many instances of the Capability ID of 10h (PCI Express Capability) are found. If more than one is found, the test terminates with a failure.
3. If the Capability ID of 10h is not found for a capability, the test terminates with a failure.
4. If a Capability ID of 10h is found for a capability, the following tests are performed:
5. Read a DWORD located at offset 0Ch (Link Capabilities register) in the PCI Express Capability structure.
6. If the device is not a Root Complex Integrated Endpoint or a Root Complex Event Collector, perform each of the following checks on the fields of the DWORD read:
 - a. For Base 1.x testing: the Max Link Speed/Supported Link Speeds field must be 0001b. All other encodings are reserved and treated as a failure.
 - b. For Base 2.x testing: the Max Link Speed/Supported Link Speeds field must be 0001b or 0010b. All other encodings are reserved and treated as a failure.
 - c. For Base 3.x testing: the Max Link Speed/Supported Link Speeds field must be 0001b, 0010b, or 0011b. All other encodings are reserved and treated as a failure.
 - d. The Maximum Link Width field must be 00 0001b, 00 0010b, 00 0100b, 00 1000b, 00 1100b, 01 0000b, or 10 0000b. All other encodings are reserved and treated as a failure.
 - e. If the ASPM Optionality Compliance field is 0, then the Active State Power Management (ASPM) Support field must be 01b or 11b. All other encodings are reserved and treated as a failure.
 - f. For Base 3.x or later testing: the ASPM Optionality Compliance field must be 1.

- g. The Port Number field:
 - i. For Switch Downstream Ports: all downstream ports in a Switch must have unique Port Numbers.
- 7. For Downstream Ports that have the Max Link Speed/Supported Link Speeds field return 0010b or greater, or that have the Hot-Plug Capable field return 1, the Data Link Layer Link Active Reporting Capable field must report 1.
- 8. For Downstream Ports that have the Max Link Speed/Supported Link Speeds field return 0010b or greater, or that have the Maximum Link Width field return greater than 000 0001b, the Link Bandwidth Notification Capability field must report 1.
- 9. Read a WORD located at offset 10h (Link Control register).
- 10. If the device is not a Root Complex Integrated Endpoint, or a Root Complex Event Collector, or a VF, the Active State Power Management (ASPM) Control field may return 00b, 01b, 10b, or 11b. Test software writes each of the allowed data values to this field (00b, 01b, 10b, and 11b) and makes sure that the same value is read back.
- 11. Read a WORD from offset 12h (Link Status register).
- 12. If the device is not a Root Complex Integrated Endpoint, or a Root Complex Event Collector, or a VF, perform each of the following checks on the fields of the WORD read:
 - a. For Base 1.x testing: the Current Link Speed field must be 0001b for 2.5 GT/s test cases. All other encodings are reserved and treated as a failure.
 - b. For Base 2.x testing: the Current Link Speed field must be 0001b for 2.5 GT/s test cases and must be 0010b for 5.0 GT/s test cases. All other encodings are reserved and treated as a failure.
 - c. For Base 3.x testing: the Current Link Speed field must be 0001b for 2.5 GT/s test cases, must be 0010b for 5.0 GT/s test cases, and must be 0011b for 8.0 GT/s test cases. All other encodings are reserved and treated as a failure.
 - d. The Current Link Speed field must be equal or less than the Max Link Speed/Supported Link Speeds field value.
 - e. The Negotiated Link Width field must be 00 0001b, 00 0010b, 00 0100b, 00 1000b, 00 1100b, 01 0000b, or 10 0000b. All other encodings are reserved and treated as a failure.
 - f. The Negotiated Link Width field must be equal or less than the Maximum Link Width field value.
 - g. For downstream ports, the Link Training field must be 0.
 - h. For downstream ports with the Data Link Layer Link Active Reporting Capable field value as 1, the Data Link Layer Link Active field must be 1.
- 13. The following register field characteristic checks are performed:

Link Capabilities Register (Offset 0Ch) — DWORD

(all except for RC Integrated Endpoint and RC Event Collector)

- | | |
|---|----|
| a. Max Link Speed/Supported Link Speeds | RO |
| b. Maximum Link Width | RO |
| c. Active State Power Management (ASPM) Support | RO |
| d. L0s Exit Latency | RO |
| e. L1 Exit Latency | RO |
| f. Clock Power Management
(Upstream Ports) | RO |

	(Downstream Ports)	RO-Zero
g.	Surprise Down Error Reporting Capable (Downstream Ports) (Upstream Ports)	RO RO-Zero
h.	Data Link Layer Link Active Reporting Capable (Downstream Ports) (Upstream Ports)	RO RO-Zero
i.	Link Bandwidth Notification Capability (Downstream Ports) (Upstream Ports)	RO RO-Zero
j.	ASPM Optionality Compliance (For Base 2.x or later testing) (for non-FLR testing) (for FLR testing)	HwInit RO-Ones
k.	RsvdP_23 (For Base 2.x or later testing)	RO-Zero
l.	RsvdP_23-22 (For Base 1.x testing)	RO-Zero
m.	Port Number (for non-FLR testing) (for FLR testing)	HwInit RO

Link Capabilities Register (Offset 0Ch) — DWORD

(RC Integrated Endpoint and RC Event Collector)

a.	RsvdP_31-0	RO-Zero
----	------------	---------

Link Control Register (Offset 10h) — WORD

(all except for RC Integrated Endpoint and RC Event Collector)

a.	Active State Power Management (ASPM) Control (for VFs) (otherwise)	RO-Zero RW
b.	RsvdP_2	RO-Zero
c.	Read Completion Boundary (RCB) (for VFs) (for Root Ports) (for Endpoints (not VFs), PCI Express to PCI/PCI-X Bridges, or PCI/PCI-X to PCI Express Bridges) (for Switch Ports)	RO-Zero RO RW or RO-Zero RO-Zero
d.	Link Disable (for Downstream Ports) (for Upstream Ports)	RW RO-Zero
e.	Retrain Link (for Downstream Ports) (for Upstream Ports)	RO-Zero RO-Zero
f.	Common Clock Configuration (for VFs) (otherwise)	RO-Zero RW

- | | | |
|----|--|----------------------------------|
| g. | Extended Synch
(for VFs)
(otherwise) | RO-Zero
RW |
| h. | Enable Clock Power Management
(for Downstream Ports)
(for Upstream Ports (not VFs) with Clock Power Management as 1)
(for Upstream Ports with Clock Power Management as 0 and VFs) | RO-Zero
RW
RO-Zero |
| i. | Hardware Autonomous Width Disable
(for Upstream Ports whose function number is non-zero and VFs)
(otherwise) | RO-Zero
RW or
RO-Zero |
| j. | Link Bandwidth Management Interrupt Enable
(for Downstream Ports with Link Bandwidth Notification Capability as 1)
(for Downstream Ports with Link Bandwidth Notification Capability as 0)
(Upstream Ports) | RW
RO-Zero
RO-Zero |
| k. | Link Autonomous Bandwidth Interrupt Enable
(for Downstream Ports with Link Bandwidth Notification Capability as 1)

(for Downstream Ports with Link Bandwidth Notification Capability as 0)

(Upstream Ports) | RW

RO-Zero

RO-Zero |
| l. | RsvdP_15-12
Note: Writing Link Disable with 1 in a Downstream Port causes the link to go down. The test should restore this bit to 0 when completing testing of this register. | RO-Zero |

Link Control Register (Offset 10h) — WORD

(RC Integrated Endpoint and RC Event Collector)

- | | | |
|----|------------|---------|
| a. | RsvdP_15-0 | RO-Zero |
|----|------------|---------|

Link Status Register (Offset 12h) — WORD

(all except for RC Integrated Endpoint and RC Event Collector and VF)

- | | | |
|----|--|----------------------------|
| a. | Current Link Speed | RO |
| b. | Negotiated Link Width | RO |
| c. | Undefined_10
Note: This was once Link Training Error. | RO |
| d. | Link Training
(for Downstream Ports)
(for Upstream Ports) | RO
RO-Zero |
| e. | Slot Clock Configuration
(for non-FLR testing)
(for FLR testing) | HwInit
RO |
| f. | Data Link Layer Link Active
(for Downstream Ports with Data Link Active Reporting Capable as 1)
(for Downstream Ports with Data Link Active Reporting Capable as 0)
(for Upstream Ports) | RO
RO-Zero
RO-Zero |
| g. | Link Bandwidth Management Status
(for Downstream Ports with Link Bandwidth Notification Capability as 1)
(for Downstream Ports with Link Bandwidth Notification Capability as 0)
(for Upstream Ports) | RW1C
RO-Zero
RO-Zero |

- | | |
|---|---------|
| h. Link Autonomous Bandwidth Status | |
| (for Downstream Ports with Link Bandwidth Notification Capability as 1) | RW1C |
| (for Downstream Ports with Link Bandwidth Notification Capability as 0) | RO-Zero |
| (for Upstream Ports) | RO-Zero |

Link Status Register (Offset 12h) — WORD

(RC Integrated Endpoint and RC Event Collector and VF)

- | | |
|---------------|---------|
| a. RsvdZ_15-0 | RO-Zero |
|---------------|---------|

14. The following default value checks are performed:

Link Capabilities Register Default Value (Offset 0Ch) — WORD

- | | |
|---------------------------------|---|
| a. ASPM Optionality Compliance | |
| (For Base 2.x or later testing) | 1 |

Link Control Register Default Value (Offset 10h) — WORD

(all except for RC Integrated Endpoint and RC Event Collector)

- | | |
|---|---|
| a. Read Completion Boundary (RCB) | |
| (for Endpoints, PCI Express to PCI/PCI-X Bridges, | |
| or PCI/PCI-X to PCI Express Bridges) | 0 |
| b. Link Disable | |
| (for Downstream Ports) | 0 |
| c. Common Clock Configuration | 0 |
| d. Extended Synch | 0 |
| e. Enable Clock Power Management | |
| (for Upstream Ports) | 0 |
| f. Hardware Autonomous Width Disable | 0 |
| g. Link Bandwidth Management Interrupt Enable | |
| (for Downstream Ports) | 0 |
| h. Link Autonomous Bandwidth Interrupt Enable | |
| (for Downstream Ports) | 0 |

15. For functions under test that have a link, the test is run at each of the following link speeds:

- | |
|---|
| a. 2.5 GT/s (for devices that are capable). See Section 2.1.2.12 for details. |
| b. 5.0 GT/s (for devices that are capable). See Section 2.1.2.13 for details. |
| c. 8.0 GT/s (for devices that are capable). See Section 2.1.2.14 for details. |

The test *fails* if:

- ☐ A PCI Express capability structure is not present.
- ☐ More than one PCI Express Capability is present.
- ☐ The Max Link Speed/Supported Link Speeds field has an invalid value.
- ☐ The Maximum Link Width field has an invalid value.
- ☐ The ASPM Optionality Compliance field is 0 and the Active State Power Management (ASPM) Support field is not 01b or 11b.
- ☐ The ASPM Optionality Compliance field is 0 (for Base 3.x or later testing only).
- ☐ For a Switch, all downstream ports do not each have a unique value in the Port Number field
- ☐ For Downstream Ports, that support greater than 5.0 GT/s or that support Hot-Plug, the Data Link Layer Link Active Reporting Capable field is 0.

- ☐ For Downstream Ports, that support greater than 2.5 GT/s or greater than x1, the Link Bandwidth Notification Capability field is 0.
- ☐ The Active State Power Management (ASPM) Control field does not accept writing all values (00b, 01b, 10b, 11b).
- ☐ The Current Link Speed field is not one of the defined values.
- ☐ The Current Link Speed field is not correct for the level of testing
- ☐ The Current Link Speed field is not equal or less than the reported Max Link Speed/Supported Link Speed field.
- ☐ The Negotiated Link Width field is not one of the defined values.
- ☐ The Negotiated Link Width field is not equal or less than the reported Maximum Link Width field.
- ☐ For a downstream port, the Link Training field is not 0.
- ☐ For a downstream port that supports Data Link Layer Link Active reporting, the Data Link Layer Active field is not 1.
- ☐ Any of the register field characteristic tests fail.
- ☐ Any of the default value tests fail.

2.2.6. TD_1_41 Link Capabilities 2, Link Control 2, and Link Status 2 Registers (PCIe Cap Ver = 2)

The test verifies that the function under test implements PCI Express Link Capabilities 2, Link Control 2, and Link Status 2 registers as defined in the relevant specifications.

Relevant Specifications

- ☐ *PCI Express Base Specification*
- ☐ *ECN PCI Express Capability Structure Expansion (to Base 1.1)*
- ☐ *Single Root I/O Virtualization and Sharing Specification*

(See Section 1.3 for additional specification revision details.)

Applicable Device/Port Types

This test is run on all device/port types.

Starting Configuration

This test is run with functions starting in the D0-Uninitialized state following the standard initialization procedures in Section 2.1.1. Testing in additional Dx states may be optionally supported. ASPM states of the upstream and downstream ports are disabled for this test.

Overview of Test Steps

Test software performs the following steps:

1. Configure the function under test following the procedure described in Section 2.1.1.

2. Examine the Capability ID field for each of the function's Capabilities. Determine how many instances of the Capability ID of 10h (PCI Express Capability) are found. If more than one is found, the test terminates with a failure.
3. If the Capability ID of 10h is not found for a capability, the test terminates with a failure.
4. If a Capability ID of 10h is found for a capability, read a WORD located at offset 02h in the PCI Express Capability Structure and if the Capability Version field is equal or less than 1h, the test terminates. (For Base 2.x or later testing, this is a failure. For Base 1.x testing, this is not a failure.)
5. If a Capability ID of 10h is found for a capability and if the Capability Version field is equal or greater than 2h, the following tests are performed:
6. For Base 3.x or later testing: read a DWORD located at offset 2Ch (Link Capabilities 2 register).
7. If the device is not a Root Complex Integrated Endpoint or a Root Complex Event Collector, perform each of the following checks on the fields of the DWORD read:
 - a. For Base 3.x testing: the Supported Link Speeds Vector field must be 000 0000b, 000 0001b, 000 0010b, 000 0011b, 000 0100b, 000 0101b, 000 0110b, or 000 0111b. All other encodings are reserved and treated as a failure.
8. For Base 2.x or later testing: read a WORD located at offset 30h (Link Control 2 register).
9. If the device is not a Root Complex Integrated Endpoint or a Root Complex Event Collector, perform each of the following checks on the fields of the WORD read:
 - a. For Base 2.x or later, if the function under test is not Function 0, then the Target Link Speed field must be 0000b.
 - b. If the function under test is Function 0 then:
 - i. For Base 2.x or later, and the Max Link Speed/Supported Link Speeds field is less than 0010b: the Target Link Speed field must be 0000b or 0001b. All other encodings are reserved and treated as a failure.
 - ii. For Base 2.x or later, and the Max Link Speed/Supported Link Speeds field is 0010b: the Target Link Speed field must be 0001b or 0010b. All other encodings are reserved and treated as a failure.
 - iii. For Base 3.x or later, and the Max Link Speed/Supported Link Speeds field is 0011b: the Target Link Speed field must be 0001b, 0010b, or 0011b. All other encodings are reserved and treated as a failure.
10. For Base 2.x or later testing: read a WORD located at offset 32h (Link Status 2 register)
11. If the device type is not a Root Complex Integrated Endpoint or a Root Complex Event Collector, or a VF, perform each of the following checks on the fields of the WORD read:
 - a. For Base 2.x or later testing: the Current De-emphasis Level field must reflect the expected level when the function under test's link is operating at 5.0 GT/s. The expected level is determined by the system specific settings for the system used to perform the test. When the function under test's link is not operating at 5.0 GT/s, or the function under test has no link, this value is not checked.
12. The following register field characteristic checks are performed:

Link Capabilities 2 Register (Offset 2Ch) — DWORD

(all except for RC Integrated Endpoint and RC Event Collector)

- a. RsvdP_0

(For Base 3.x or later testing)

RO-Zero

- b. Supported Link Speeds Vector
(For Base 3.x or later testing) RO
- c. Crosslink Supported
(For Base 3.x or later testing) RO
- d. RsvdP_31-9
(For Base 3.x or later testing) RO-Zero
- e. RsvdP_31-0
(For Base 1.x or Base 2.x testing) RO-Zero

Link Capabilities 2 Register (Offset 2Ch) — DWORD

(RC Integrated Endpoint and RC Event Collector)

- a. RsvdP_31-0 RO-Zero

Link Control 2 Register (Offset 30h) — WORD

(all except for RC Integrated Endpoint and RC Event Collector)

- a. Target Link Speed
(For Base 2.x or later testing)
(for Upstream Ports whose function number is non-zero) RO-Zero
(otherwise, if Max Link Speed/Supported Link Speeds is 0001b) RWS or
RO-Zero
(otherwise) RWS

Note: This test must only write supported values as reported in the Max Link Speed/Supported Link Speeds field.

- b. Enter Compliance
(For Base 2.x or later testing)
(for Upstream Ports whose function number is non-zero) RO-Zero
(otherwise, if Max Link Speed/Supported Link Speeds is 0001b) RWS or
RO-Zero
(otherwise, if Max Link Speed/Supported Link Speeds is 0010b or greater) RWS

Note: Enter Compliance field cannot be tested for stickiness even though register field attribute is RWS because this field is reset to zero (upon exiting Polling.Compliance) during successful link training.

Note: Enter Compliance field cannot be tested for default value, as setting it to 1 and then doing a Hot Reset or DL_Down (link disable/enable), will cause the link to go to Compliance Pattern state and it may not return to L0 again.

- c. Hardware Autonomous Speed Disable
(For Base 2.x or later testing)
(for Upstream Ports whose function number is non-zero) RO-Zero
(otherwise) RWS or
RO-Zero
- d. Selectable De-emphasis
(For Base 2.x or later testing)
(for Downstream Ports, for non-FLR testing) HwInit
(for Downstream Ports, for FLR testing) RO
(for Upstream Ports) RO-Zero

- e. Transmit Margin
 (For Base 2.x or later testing)
 (for Upstream Ports whose function number is non-zero) RO-Zero
 (otherwise, if Max Link Speed/Supported Link Speeds is 0001b) RWS or
 RO-Zero
 (otherwise, if Max Link Speed/Supported Link Speeds is 0010b or greater) RWS
 Note: The Transmit margin field cannot be tested for stickiness even though register field attribute is RWS because this field is reset to zero (in Polling.Configuration) during successful link training.
- f. Enter Modified Compliance
 (For Base 2.x or later testing)
 (for Upstream Ports whose function number is non-zero) RO-Zero
 (otherwise, if Max Link Speed/Supported Link Speeds is 0001b) RWS or
 RO-Zero
 (otherwise, if Max Link Speed/Supported Link Speeds is 0010b or greater) RWS
- g. Compliance SOS
 (For Base 2.x or later testing)
 (for Upstream Ports whose function number is non-zero) RO-Zero
 (otherwise, if Max Link Speed/Supported Link Speeds is 0001b) RWS or
 RO-Zero
 (otherwise, if Max Link Speed/Supported Link Speeds is 0010b or greater) RWS
- h. Compliance Preset/De-emphasis
 (For Base 3.x or later testing)
 (for Upstream Ports whose function number is non-zero) RO-Zero
 (otherwise, if Max Link Speed/Supported Link Speeds is 0001b) RWS or
 RO-Zero
 (otherwise, if Max Link Speed/Supported Link Speeds is 0010b or greater) RWS
 Note: For Base 3.x or later, this field consists of bits 15-12 of this register.
- i. Compliance De-emphasis
 (For Base 2.x testing)
 (for Upstream Ports whose function number is non-zero) RO-Zero
 (otherwise, if Max Link Speed/Supported Link Speeds is 0001b) RWS or
 RO-Zero
 (otherwise, if Max Link Speed/Supported Link Speeds is 0010b or greater) RWS
 Note: For Base 2.x, this field consists of bit 12 of this register.
- j. RsvdP_15-13
 (For Base 2.x testing) RO-Zero
- k. RsvdP_15-0
 (For Base 1.x testing) RO-Zero
- Link Control 2 Register (Offset 30h) — WORD**
 (RC Integrated Endpoint and RC Event Collector)
- a. RsvdP_15-0 RO-Zero

Link Status 2 Register (Offset 32h) — WORD

(all except for RC Integrated Endpoint and RC Event Collector)

- a. Current De-emphasis Level
(For Base 2.x or later testing)
(for VFs) RO-Zero
(for non-VFs) RO
- b. Equalization Complete
(For Base 3.x or later testing)
(for VFs) RO-Zero
(for non-VFs whose function number is non-zero) RO-Zero
(for non-VFs whose function number is zero) ROS
Note: The Equalization Complete field cannot be tested for stickiness even though the register field attribute is ROS because this bit is reset to zero on link down, and is then again updated during link equalization.
- c. Equalization Phase 1 Successful
(For Base 3.x or later testing)
(for VFs) RO-Zero
(for non-VFs whose function number is non-zero) RO-Zero
(for non-VFs whose function number is zero) ROS
Note: The Equalization Phase 1 Successful field cannot be tested for stickiness even though the register field attribute is ROS because this bit is reset to zero on link down, and is then again updated during link equalization.
- d. Equalization Phase 2 Successful
(For Base 3.x or later testing)
(for VFs) RO-Zero
(for non-VFs whose function number is non-zero) RO-Zero
(for non-VFs whose function number is zero) ROS
Note: The Equalization Phase 2 Successful field cannot be tested for stickiness even though the register field attribute is ROS because this bit is reset to zero on link down, and is then again updated during link equalization.
- e. Equalization Phase 3 Successful
(For Base 3.x or earlier)
(for VFs) RO-Zero
(for non-VFs whose function number is non-zero) RO-Zero
(for non-VFs whose function number is zero) ROS
Note: The Equalization Phase 3 Successful field cannot be tested for stickiness even though the register field attribute is ROS because this bit is reset to zero on link down, and is then again updated during link equalization.
- f. Link Equalization Request
(For Base 3.x or later testing)
(for VFs) RO-Zero
(for non-VFs whose function number is non-zero) RO-Zero
(for non-VFs whose function number is zero) RW1CS
Note: The Link Equalization Request field cannot be tested for stickiness even though the register field attribute is RW1CS because this bit is updated during link equalization.
- g. RsvdZ_15-6
(For Base 3.x or later testing) RO-Zero

- | | |
|---|---------|
| h. RsvdZ_15-1
(For Base 2.x testing) | RO-Zero |
| i. RsvdZ_15-0
(For Base 1.x testing) | RO-Zero |

Link Status 2 Register (Offset 32h) — WORD

(RC Integrated Endpoint and RC Event Collector)

- | | |
|---------------|---------|
| a. RsvdZ_15-0 | RO-Zero |
|---------------|---------|

13. The following default value checks are performed:

Link Control 2 Register Default Value (Offset 30h) — WORD

(all except for RC Integrated Endpoint and RC Event Collector)

- | | |
|--|---|
| a. Target Link Speed
(For Base 2.x or later testing)
(all single function devices)

(function 0 of a multi-function device)

(not function 0 of a multi-function device)
Note: The Highest supported speed value [HLS] is the value returned in the Max Link Speed/Supported Link Speeds field (Link Control register). [HLS] may also be 0000b if the Max Link Speed/Supported Link Speeds field returns 0001b (2.5 GT/s only).
Note: This test must only write supported values as reported in the Max Link Speed/Supported Link Speeds field. | [HLS]
see Note below
[HLS]
see Note below
0 |
| b. Hardware Autonomous Speed Disable
(For Base 2.x or later testing) | 0 |
| c. Transmit Margin
(For Base 2.x or later testing) | 000b |
| d. Enter Modified Compliance
(For Base 2.x or later testing) | 0 |
| e. Compliance SOS
(For Base 2.x or later testing) | 0 |
| f. Compliance Preset/De-emphasis
(For Base 3.x or later testing)
Note: For Base 3.x or later, this field consists of bits 15-12 of this register. | 0000b |
| g. Compliance De-emphasis
(For Base 2.x testing)
Note: For Base 2.x, this field consists of bit 12 of this register. | 0 |

Link Status 2 Register Default Value (Offset 32h) — WORD

(all except for RC Integrated Endpoint and RC Event Collector)

- | | |
|---|---|
| a. Link Equalization Request
(For Base 3.x or later testing:
if Max Link Speed/Supported Link Speeds is 0010b or less) | 0 |
| Note: A function under test that supports 8.0 GT/s will participate in Link Equalization, so the default values of the equalization status fields cannot be tested on an 8.0 GT/s capable device. | |

14. For functions under test that have a link, the test is run at each of the following link speeds:
- 2.5 GT/s (for devices that are capable). See Section 2.1.2.12 for details.
 - 5.0 GT/s (for devices that are capable). See Section 2.1.2.13 for details.
 - 8.0 GT/s (for devices that are capable). See Section 2.1.2.14 for details.

The test *fails* if:

- ☐ A PCI Express capability structure is not present.
- ☐ More than one PCI Express Capability is present.
- ☐ The Supported Link Speeds Vector field is not one of the allowed values.
- ☐ The Target Link Speed field is not one of the allowed values.
- ☐ For Function 0 only, the Target Link Speed field is not the highest supported link speed.
- ☐ When operating at 5.0 GT/s, the Current De-emphasis Level field does not match the expected setting of the specific test system.
- ☐ Any of the register field characteristic tests fail.
- ☐ Any of the default value tests fail.

2.2.7. TD_1_49 Slot Capabilities, Slot Control, and Slot Status Registers (PCIe Cap Ver = 2)

The test verifies that if the function under test reports support for a Slot Connector, it implements the Slot Capabilities, Slot Control, and Slot Status registers as defined in the relevant specifications. If the Slot Implemented field is 0 the Slot Capabilities, Slot Control, and Slot Status registers must be hardwired to zero (except for the Presence Detect State field in the Slot Status register which must be hardwired to 1 only for downstream ports).

Relevant Specifications

- ☐ *PCI Express Base Specification*

(See Section 1.3 for additional specification revision details.)

Applicable Device/Port Types

This test is run on all device/port types.

Starting Configuration

This test is run with functions starting in the D0-Uninitialized state following the standard initialization procedures in Section 2.1.1. Testing in additional Dx states may be optionally supported. ASPM states of the upstream and downstream ports are disabled for this test.

Overview of Test Steps

Test software performs the following steps:

1. Configure the function under test following the procedure described in Section 2.1.1.
2. Examine the Capability ID field for each of the function's Capabilities. Determine how many instances of the Capability ID of 10h (PCI Express Capability) are found. If more than one is found, the test terminates with a failure.
3. If the Capability ID of 10h is not found for a capability, the test terminates with a failure.

4. If a Capability ID of 10h is found for a capability, read a WORD located at offset 02h in the PCI Express Capability Structure and if the Capability Version field is equal or less than 1h, the test terminates (this is not a failure and the test reports skipped).
5. If a Capability ID of 10h is found for a capability and if the Capability Version field is equal or greater than 2h, the following tests are performed:
6. If the Slot Implemented field is 0, then skip to step 11.
7. Read the DWORD located at offset 14h (Slot Capabilities register) in the PCI Express Capability Structure.
8. If the Attention Indicator Present field returns 1, then the following tests are performed:
 - a. Test software writes the Attention Indicator Control field with 01b.
 - b. If the No Command Completed Support field (Slot Capabilities register) is 1, test software polls the Command Completed field (Slot Status register) until it reads back a 1. If this does not occur after 1 second of polling, then step a) is repeated. If this still does not occur after 10 repeats, this part of the test terminates with a test failure and the test continues at step 9. When the Command Completed field (Slot Status register) returns 1, test software writes the Command Completed field (Slot Status register) with 1 (to clear the status).
 - c. Test software reads the Attention Indicator Control field and checks that it returns the value previously written. If not, then this part of the test terminates with a test failure and the test continues at step 9.
 - d. Repeat steps a-c using the Attention Indicator Control field data values as follows: 10b; 11b.
 - e. Perform a default value check on the Attention Indicator Control field and confirm that it returns one of the allowed default values (01b, 10b, or 11b).
9. If the Power Indicator Present field returns 1, then the following tests are performed:
 - a. Test software writes the Power Indicator Control field with 01b.
 - b. If the No Command Completed Support field (Slot Capabilities register) is 1, test software polls the Command Completed field (Slot Status register) until it reads back a 1. If this does not occur after 1 second of polling, then step a) is repeated. If this still does not occur after 10 repeats, this part of the test terminates with a test failure and the test continues at step 10. When the Command Completed field (Slot Status register) returns 1, test software writes the Command Completed field (Slot Status register) with 1 (to clear the status).
 - c. Test software reads the Power Indicator Control field and checks that it returns the value previously written. If not, then this part of the test terminates with a test failure and the test continues at step 10.
 - d. Repeat steps a-c using the Power Indicator Control field data values as follows: 10b; 11b.
 - e. Perform a default value check on the Power Indicator Control field and confirm that it returns one of the allowed default values (01b, 10b, or 11b).
10. If the Power Controller Implemented field returns 1, then the following tests are performed:
 - a. Test software writes the Power Controller Control field with 0.
 - b. If the No Command Completed Support field (Slot Capabilities register) is 1, test software polls the Command Completed field (Slot Status register) until it reads back a 0. If this does not occur after 1 second of polling, then step a) is repeated. If this still does not occur after 10 repeats, this part of the test terminates with a test failure and the test continues at step 11. When the Command Completed field (Slot Status register) returns 1, test software writes the Command Completed field (Slot Status register) with 1 (to clear the status).

- c. Test software reads the Power Controller Control field and checks that it returns the value previously written. If not, this part of the test terminates with a test failure and the test continues at step 11.
- d. Test software reads the Power Fault Detected field (Slot Status register) and checks that it returns 0. If not, then this part of the test terminates with a test failure and the test continues at step 11.
(Note: Power Controller Control field data value 1 is not tested, as this would power down the hot-plug slot. There is no intention in this test to remove power from any card in any slot.)

11. The following register field characteristic checks are performed:

Slot Capabilities Register (Offset 14h) — DWORD

(for Downstream Ports with Slot Implemented as 1)

- | | |
|--|--------------|
| a. Attention Button Present
(for non-FLR testing)
(for FLR testing) | HwInit
RO |
| b. Power Controller Present
(for non-FLR testing)
(for FLR testing) | HwInit
RO |
| c. MRL Sensor Present
(for non-FLR testing)
(for FLR testing) | HwInit
RO |
| d. Attention Indicator Present
(for non-FLR testing)
(for FLR testing) | HwInit
RO |
| e. Power Indicator Present
(for non-FLR testing)
(for FLR testing) | HwInit
RO |
| f. Hot-Plug Surprise
(for non-FLR testing)
(for FLR testing) | HwInit
RO |
| g. Hot-Plug Capable
(for non-FLR testing)
(for FLR testing) | HwInit
RO |
| h. Slot Power Limit Value
(for non-FLR testing)
(for FLR testing) | HwInit
RO |
| i. Slot Power Limit Scale
(for non-FLR testing)
(for FLR testing) | HwInit
RO |
| j. Electromechanical Interlock Present
(for non-FLR testing)
(for FLR testing) | HwInit
RO |
| k. No Command Completed Support
(for non-FLR testing)
(for FLR testing) | HwInit
RO |
| l. Physical Slot Number
(for non-FLR testing)
(for FLR testing) | HwInit
RO |

Slot Capabilities Register (Offset 14h) — DWORD

(for Downstream Ports with Slot Implemented as 0, and all functions other than Downstream Ports)

- | | |
|---------------|---------|
| a. RsvdP_31-0 | RO-Zero |
|---------------|---------|

Slot Control Register (Offset 18h) — WORD

(for Downstream Ports with Slot Implemented as 1)

- | | |
|---|------------------------|
| a. Attention Button Pressed Enable
(if Attention Button Present is 1)
(if Attention Button Present is 0) | RW
RW or
RO-Zero |
| b. Power Fault Detected Enable | RW or
RO-Zero |
| c. MRL Sensor Changed Enable
(if MRL Sensor Present is 1)
(if MRL Sensor Present is 0) | RW
RW or
RO-Zero |
| d. Presence Detect Changed Enable
(if Hot-Plug Capable is 1)
(if Hot-Plug Capable is 0) | RW
RW or
RO-Zero |
| e. Command Completed Interrupt Enable
(if No Command Completed Support is 0)
(if No Command Completed Support is 1) | RW
RW or
RO-Zero |
| f. Hot-Plug Interrupt Enable
(if Hot-Plug Capable is 1)
(if Hot-Plug Capable is 0) | RW
RW or
RO-Zero |
| g. Attention Indicator Control
(if Attention Indicator Present is 0) | RW or
RO-Zero |
| h. Power Indicator Control
(if Power Indicator Present is 0) | RW or
RO-Zero |
| i. Power Controller Control
(if Power Controller Present is 0) | RW or
RO
RO-Zero |
| j. Electromechanical Interlock Control | RO-Zero |
| k. Data Link Layer State Changed Enable
(if Data Link Layer Link Active Reporting Capable is 1)
(if Data Link Layer Link Active Reporting Capable is 0) | RW
RW or
RO-Zero |
| l. RsvdP_15-13 | RO-Zero |

Slot Control Register (Offset 18h) — WORD

(for Downstream Ports with Slot Implemented as 0, and all functions other than Downstream Ports)

- | | |
|---------------|---------|
| a. RsvdP_15-0 | RO-Zero |
|---------------|---------|

Slot Status Register (Offset 1Ah) — WORD

(for Downstream Ports with Slot Implemented as 1)

- | | |
|---|---------|
| a. Attention Button Pressed
(if Attention Button Present is 1) | RW1C |
| (if Attention Button Present is 0) | RO-Zero |
| b. Power Fault Detected
(if Power Controller Present is 1) | RW1C |
| (if Power Controller Present is 0) | RO-Zero |
| c. MRL Sensor Changed
(if MRL Sensor Present is 1) | RW1C |
| (if MRL Sensor Present is 0) | RO-Zero |
| d. Presence Detect Changed | RW1C |
| e. Command Completed
(if No Command Completed Support is 0) | RW1C |
| (if No Command Completed Support is 1) | RO-Zero |
| f. MRL Sensor State | RO |
| g. Presence Detect State | RO |
| h. Electromechanical Interlock Status | RO |
| i. Data Link Layer State Changed | RW1C |
| j. RsvdZ_15-9 | RO-Zero |

Slot Status Register (Offset 1Ah) — WORD

(for Downstream Ports with Slot Implemented as 0)

- | | |
|--------------------------|---------|
| a. RsvdZ_5-0 | RO-Zero |
| b. Presence Detect State | RO-Ones |
| c. RsvdZ_15-7 | RO-Zero |

Slot Status Register (Offset 1Ah) — WORD

(for all functions other than Downstream Ports)

- | | |
|---------------|---------|
| a. RsvdZ_15-0 | RO-Zero |
|---------------|---------|

12. The following default value checks are performed:

Slot Control Register Default Value (Offset 18h) — WORD

(for Downstream Ports with Slot Implemented as 1)

- | | |
|---|---|
| a. Attention Button Pressed Enable | 0 |
| b. Power Fault Detected Enable | 0 |
| c. MRL Sensor Changed Enable | 0 |
| d. Presence Detect Changed Enable | 0 |
| e. Command Completed Interrupt Enable | 0 |
| f. Hot-Plug Interrupt Enable | 0 |
| g. Data Link Layer State Changed Enable | 0 |

Slot Status Register Default Value (Offset 1Ah) — WORD

(for Downstream Ports with Slot Implemented as 1)

a. Attention Button Pressed	0
b. Power Fault Detected	0
c. MRL Sensor Changed	0
d. Presence Detect Changed	0
e. Command Completed	0

13. For functions under test that have a link, the test is run at each of the following link speeds:
- a. 2.5 GT/s (for devices that are capable). See Section 2.1.2.12 for details.
 - b. 5.0 GT/s (for devices that are capable). See Section 2.1.2.13 for details.
 - c. 8.0 GT/s (for devices that are capable). See Section 2.1.2.14 for details.

The test *fails* if:

- ☐ A PCI Express capability structure is not present.
- ☐ More than one PCI Express Capability is present.
- ☐ The Attention Indicator Present field is 1, but the Power Indicator Control field cannot be written with one of the valid values, or the Command Completed field status does not return 1.
- ☐ The Power Indicator Present field is 1, but the Power Indicator Control field cannot be written with one of the valid values, or the Command Completed field status does not return 1.
- ☐ The Power Controller Implemented field is 1, but the Power Controller Control field cannot be written with the ON value (0), or the Command Completed field status is not returned, or the Power Fault Detected field status returns 1.
- ☐ Any of the register field characteristic tests fail.
- ☐ Any of the default value tests fail.

2.2.8. TD_1_50 Slot Capabilities 2, Slot Control 2, and Slot Status 2 Registers (PCIe Cap Ver = 2)

The test verifies that if the function under test reports support for a Slot Connector, it implements the Slot Capabilities 2, Slot Control 2, and Slot Status 2 registers as defined in the relevant specifications. If the Slot Implemented field is 0 the Slot Capabilities 2, Slot Control 2, and Slot Status 2 registers must be hardwired to zero.

Relevant Specifications

- ☐ *PCI Express Base Specification*
- ☐ *ECN PCI Express Capability Structure Expansion (to Base 1.1)*

(See Section 1.3 for additional specification revision details.)

Applicable Device/Port Types

This test is run on all device/port types.

Starting Configuration

This test is run with functions starting in the D0-Uninitialized state following the standard initialization procedures in Section 2.1.1. Testing in additional Dx states may be optionally supported. ASPM states of the upstream and downstream ports are disabled for this test.

Overview of Test Steps

Test software performs the following steps:

1. Configure the function under test following the procedure described in Section 2.1.1.
2. Examine the Capability ID field for each of the function's Capabilities. Determine how many instances of the Capability ID of 10h (PCI Express Capability) are found. If more than one is found, the test terminates with a failure.
3. If the Capability ID of 10h is not found for a capability, the test terminates with a failure.
4. If a Capability ID of 10h is found for a capability, read a WORD located at offset 02h in the PCI Express Capability Structure and if the Capability Version field is equal or less than 1h, the test terminates. For Base 2.x or later testing, this is a failure. For Base 1.x testing, this is not a failure, and the test result is reported as skipped.
5. If a Capability ID of 10h is found for a capability and if the Capability Version field is equal or greater than 2h, the following register field characteristic checks are performed:

Slot Capabilities 2 Register (Offset 34h) — DWORD

- | | |
|---------------|---------|
| a. RsvdP_31-0 | RO-Zero |
|---------------|---------|

Slot Control 2 Register (Offset 38h) — WORD

- | | |
|---------------|---------|
| a. RsvdP_15-0 | RO-Zero |
|---------------|---------|

Slot Status 2 Register (Offset 3Ah) — WORD

- | | |
|---------------|---------|
| a. RsvdZ_15-0 | RO-Zero |
|---------------|---------|

6. For functions under test that have a link, the test is run at each of the following link speeds:
 - a. 2.5 GT/s (for devices that are capable). See Section 2.1.2.12 for details.
 - b. 5.0 GT/s (for devices that are capable). See Section 2.1.2.13 for details.
 - c. 8.0 GT/s (for devices that are capable). See Section 2.1.2.14 for details.

The test *fails* if:

- ☐ A PCI Express capability structure is not present.
- ☐ More than one PCI Express Capability is present.
- ☐ Any of the register field characteristic tests fail.

2.2.9. TD_1_51 Root Capabilities, Root Control, and Root Status Registers (PCIe Cap Ver = 2)

The test verifies that if the function under test is a Root Port or a Root Complex Event Collector, it implements the Root Capabilities, Root Control, and Root Status registers as defined in the relevant specifications. For other device types the Root Capabilities, Root Control, and Root Status registers must be hardwired to zero.

Relevant Specifications

- ☐ *PCI Express Base Specification*

(See Section 1.3 for additional specification revision details.)

Applicable Device/Port Types

This test is run on all device/port types.

Starting Configuration

This test is run with functions starting in the D0-Uninitialized state following the standard initialization procedures in Section 2.1.1. Testing in additional Dx states may be optionally supported. ASPM states of the upstream and downstream ports are disabled for this test.

Overview of Test Steps

Test software performs the following steps:

1. Configure the function under test following the procedure described in Section 2.1.1.
2. Examine the Capability ID field for each of the function's Capabilities. Determine how many instances of the Capability ID of 10h (PCI Express Capability) are found. If more than one is found, the test terminates with a failure.
3. If the Capability ID of 10h is not found for a capability, the test terminates with a failure.
4. If a Capability ID of 10h is found for a capability, read a WORD located at offset 02h in the PCI Express Capability Structure and if the Capability Version field is equal or less than 1h, the test terminates (this is not a failure and the test reports skipped).
5. If a Capability ID of 10h is found for a capability and if the Capability Version field is equal or greater than 2h, the following register field characteristic checks are performed:

Root Control Register (Offset 1Ch) — WORD

(Root Port or RC Event Collector)

- | | |
|--|---------|
| a. System Error on Correctable Error Enable | RW |
| b. System Error on Non-Fatal Error Enable | RW |
| c. System Error on Fatal Error Enable | RW |
| d. PME Interrupt Enable | RW |
| e. CRS Software Visibility Enable | |
| (for Root Ports with CRS Software Visibility as 1) | RW |
| (for Root Ports with CRS Software Visibility as 0) | RO-Zero |
| (for RC Event Collectors) | RO-Zero |
| f. RsvdP_15-5 | RO-Zero |

Root Control Register (Offset 1Ch) — WORD

(for all except Root Port and RC Event Collector)

- | | |
|---------------|---------|
| a. RsvdP_15-0 | RO-Zero |
|---------------|---------|

Root Capabilities Register (Offset 1Eh) — WORD

(Root Port or RC Event Collector)

- | | |
|----------------------------|---------|
| a. CRS Software Visibility | |
| (for Root Ports) | RO |
| (for RC Event Collectors) | RO-Zero |
| b. RsvdP_15-1 | RO-Zero |

Root Capabilities Register (Offset 1Eh) — WORD

(for all except Root Port and RC Event Collector)

- | | |
|---------------|---------|
| a. RsvdP_15-0 | RO-Zero |
|---------------|---------|

Root Status Register (Offset 20h) — DWORD

(Root Port or RC Event Collector)

- | | |
|---------------------|---------|
| a. PME Requester ID | RO |
| b. PME Status | RW1C |
| c. PME Pending | RO |
| d. RsvdZ_31-18 | RO-Zero |

Root Status Register (Offset 20h) — DWORD

(for all except Root Port and RC Event Collector)

- | | |
|---------------|---------|
| a. RsvdZ_31-0 | RO-Zero |
|---------------|---------|

6. If a Capability ID of 10h is found for a capability and if the Capability Version field is equal or greater than 2h, the following default value checks are performed:

Root Control Register (Offset 1Ch) — WORD

(Root Port or RC Event Collector)

- | | |
|---|---|
| a. System Error on Correctable Error Enable | 0 |
| b. System Error on Non-Fatal Error Enable | 0 |
| c. System Error on Fatal Error Enable | 0 |
| d. PME Interrupt Enable | 0 |
| e. CRS Software Visibility Enable | |
| (for Root Ports) | 0 |

Root Status Register (Offset 20h) — DWORD

(Root Port or RC Event Collector)

- | | |
|---------------|---|
| a. PME Status | 0 |
|---------------|---|

7. For functions under test that have a link, the test is run at each of the following link speeds:
- 2.5 GT/s (for devices that are capable). See Section 2.1.2.12 for details.
 - 5.0 GT/s (for devices that are capable). See Section 2.1.2.13 for details.
 - 8.0 GT/s (for devices that are capable). See Section 2.1.2.14 for details.

The test *fails* if:

- ☐ A PCI Express capability structure is not present.
- ☐ More than one PCI Express Capability is present.
- ☐ Any of the register field characteristic tests fail.
- ☐ Any of the default value tests fail.

2.2.10. TD_1_6 MSI Capability Structure

The test verifies that if the function under test reports a MSI capability structure, it is implemented as defined in the relevant specifications. (If legacy interrupt support is indicated, the function must implement at least either a MSI or a MSI-X capability structure. The MSI-X capability structure is tested elsewhere.)

Relevant Specifications

- ❑ *PCI Express Base Specification*
- ❑ *PCI Local Bus Specification, Revision 3.0*
- ❑ *Single Root I/O Virtualization and Sharing Specification*

(See Section 1.3 for additional specification revision details.)

Applicable Device/Port Types

This test is run on all device/port types.

Starting Configuration

This test is run with functions starting in the D0-Uninitialized state following the standard initialization procedures in Section 2.1.1. Testing in additional Dx states may be optionally supported. ASPM states of the upstream and downstream ports are disabled for this test.

Overview of Test Steps

Test software performs the following steps:

1. Configure the function under test following the procedure described in Section 2.1.1.
2. A byte is read from location 3Dh (Interrupt Pin register) in configuration space for the function under test.
3. If the Interrupt Pin register value is non-zero, the function must support at least either MSI or MSI-X interrupts and must implement the necessary MSI or MSI-X capability structure (only the MSI Capability structure will be tested here).
4. Examine the Capability ID field for each of the function's Capabilities. Determine how many instances of the Capability ID of 05h (MSI Capability) are found. If more than one is found, the test terminates with a failure.
5. If a Capability ID of 05h is found for a capability, the following checks are performed on the MSI capability structure.
6. A WORD is read from offset 00h in the MSI capability structure (Capability ID field, Next Capability Pointer field)
 - a. The Next Capability Pointer field must be 00h or greater than 3Fh and the lower 2 bits of this field must be 00b.
7. A WORD is read from offset 02h in the MSI capability structure. (MSI Control register)
 - a. The Multiple Message Capable field must not return 110b or 111b (Reserved).

8. Test software writes each supported data value (based on the value read from the Multiple Message Capable field) into the Multiple Message Enable field (e.g., if the Multiple Message Capable field reports 001b meaning two messages are supported, then data values of 001b and 000b are written to the Multiple Message Enable field). After each write test software checks that the same value is read back from the Multiple Message Enable field.
9. The following register field characteristic checks are performed:

MSI Capability Header (Offset 00h) — WORD

- | | |
|----------------------------|----|
| a. Capability ID | RO |
| b. Next Capability Pointer | RO |

Message Control Register (Offset 02h) — WORD

- | | |
|---|---------------|
| a. MSI Enable | RW |
| b. Multiple Message Capable | RO |
| c. Multiple Message Enable | RW |
| d. 64 Bit Address Capable | RO |
| e. Per-vector Masking Capable
(for PFs and VFs)
(otherwise) | RO-Ones
RO |
| f. Reserved_15-9 | RO-Zero |

Message Address Register (Offset 04h) — DWORD

- | | |
|--------------------|---------|
| a. Reserved_1-0 | RO-Zero |
| b. Message Address | RW |

Message Upper Address Register (Offset 08h) — DWORD (Only if 64 Bit Address Capable is 1)

- | | |
|--------------------------|----|
| a. Message Upper Address | RW |
|--------------------------|----|

Message Data Register (Offset 08h if 64 Bit Address Capable is 0 or Offset 0Ch if 64 Bit Address Capable is 1) — WORD

- | | |
|-----------------|----|
| a. Message Data | RW |
|-----------------|----|

Reserved Register (Offset 0Ah if 64 Bit Address Capable is 0 or Offset 0Eh if 64 Bit Address Capable is 1) — WORD (Only if Per-Vector Masking Capable is 1)

- | | |
|------------------|---------|
| a. Reserved_15-0 | RO-Zero |
|------------------|---------|

Mask Bits Register (Offset 0Ch if 64 Bit Address Capable is 0 or 10h if 64 Bit Address Capable is 1) — DWORD (Only if Per-Vector Masking Capable is 1)

The value returned in the Multiple Message Capable field is used to determine N as follows: $N = 2^{**} [\text{value}]$.

- | | |
|--------------------|---------|
| a. (bits N-1 to 0) | RW |
| b. (bits 31 to N) | RO-Zero |

Pending Bits Register (Offset 10h if 64 Bit Address Capable is 0 or Offset 14h if 64 Bit Address Capable is 1) — DWORD (Only if Per-Vector Masking Capable is 1)

The value returned in the Multiple Message Capable field is used to determine N as follows: $N = 2^{**} [\text{value}]$.

- | | |
|--------------------|---------|
| a. (bits N-1 to 0) | RO |
| b. (bits 31 to N) | RO-Zero |

10. The following default value checks are performed:

Message Control Register Default Value (Offset 02h) — WORD

- | | |
|----------------------------|------|
| a. MSI Enable | 0 |
| b. Multiple Message Enable | 000b |

Mask Bits Register Default Value (Offset 0Ch if 64 Bit Address Capable is 0 or 10h if 64 Bit Address Capable is 1) — DWORD (Only if Per-Vector Masking Capable is 1)

The value returned in the Multiple Message Capable field is used to determine N as follows: $N = 2^{**} [\text{value}]$.

- | | |
|------------------------|--------------|
| a. Mask Bits (N-1 : 0) | 0 (each bit) |
|------------------------|--------------|

Pending Bits Register Default Value (Offset 10h if 64 Bit Address Capable is 0 or Offset 14h if 64 Bit Address Capable is 1) — DWORD (Only if Per-Vector Masking Capable is 1)

The value returned in the Multiple Message Capable field is used to determine N as follows: $N = 2^{**} [\text{value}]$.

- | | |
|---------------------------|--------------|
| a. Pending Bits (N-1 : 0) | 0 (each bit) |
|---------------------------|--------------|

11. For functions under test that have a link, the test is run at each of the following link speeds:

- a. 2.5 GT/s (for devices that are capable). See Section 2.1.2.12 for details.
- b. 5.0 GT/s (for devices that are capable). See Section 2.1.2.13 for details.
- c. 8.0 GT/s (for devices that are capable). See Section 2.1.2.14 for details.

The test *fails* if:

- ☐ More than one MSI capability structure is present.
- ☐ A non-zero Next Capability Pointer field value is less than or equal to 3Fh, or the lower 2 bits are non-zero.
- ☐ The Multiple Message Capable field reports a reserved value.
- ☐ Any of the allowed multiple message values cannot be written to and read back from the Multiple Message Enable field.
- ☐ Any of the register field characteristic tests fail.
- ☐ Any of the default value tests fail.

2.2.11. TD_1_7 Advanced Error Reporting Extended Capability Structure

The test verifies that if the function under test reports a PCI Express Advanced Error Reporting (AER) Capability structure it is implemented as defined in the relevant specifications.

(Note: In the various revisions of the specification, the default value of unimplemented Uncorrectable Error Severity register bits was defined in a contradictory manner. The different revisions attempted to correct this with only limited success. The current specification definition makes the default value of unimplemented Uncorrectable Error Severity register bits implementation-specific and therefore having no testable value. This is the definition that this test document implements regardless of the revision testing level, since it can be supported by all specification revisions.)

Relevant Specifications

- ❑ *PCI Express Base Specification*
- ❑ *ECN Access Control Services (to Base 1.1)*
- ❑ *ECN Internal Error Reporting (to Base 2.0)*
- ❑ *ECN Multicast (to Base 2.0)*
- ❑ *ECN Atomic Operations (to Base 2.0)*
- ❑ *ECN TLP Prefix (to Base 2.0)*
- ❑ *PCI Express to PCI/PCI-X Bridge Specification, Revision 1.0*
- ❑ *Single Root I/O Virtualization and Sharing Specification*

(See Section 1.3 for additional specification revision details.)

Applicable Device/Port Types

This test is run on all device/port types.

This test is run on any RCRB associated with the function under test.

Starting Configuration

This test is run with functions starting in the D0-Uninitialized state following the standard initialization procedures in Section 2.1.1. Testing in additional Dx states may be optionally supported. ASPM states of the upstream and downstream ports are disabled for this test.

Overview of Test Steps

Test software performs the following steps:

1. Configure the function under test following the procedure described in Section 2.1.1.
2. Examine the Extended Capability ID field for each of the detected Extended Capabilities. Determine how many instances of the Extended Capability ID of 0001h (Advanced Error Reporting Extended Capability) are found. If more than one is found, the test terminates with a failure.
3. If the Extended Capability ID of 0001h is found in an RCRB, the test terminates with a failure.
4. If an Extended Capability ID of 0001h is found for an extended capability the following checks are performed on that extended capability structure:
5. DWORDs are read from offsets 00h to 28h to obtain the default values for each of the capability fields. Root Ports and Root Complex Event Collectors, that report AER Capability Version of 1h, additionally read offsets 2Ch to 34h to obtain those default values. Any non-Bridge function that reports AER Capability Version of 2h or greater additionally read offsets 2Ch to 44h to obtain these default values. Any Bridges additionally read offsets 2Ch to 48h to obtain these default values.
6. For Base 1.x testing: the AER Capability Version field must be 1h.
7. For Base 2.x or later testing, if the function under test reports the Capability Version field (PCI Express Capabilities register) is 1h or less:
 - a. The AER Capability Version field can be 1h or 2h.

8. For Base 2.x or later testing, if the function under test reports the Capability Version field (PCI Express Capabilities register) is 2h or greater:
 - a. If the End-End TLP Prefix Supported field (Device Capabilities 2 register) is 0, then the AER Capability Version field can be 1h or 2h.
9. For Base 2.x or later testing, if the function under test reports the Capability Version field (PCI Express Capabilities register) is 2h or greater:
 - a. If the End-End TLP Prefix Supported field (Device Capabilities 2 register) is 1, then the AER Capability Version field must be 2h.
10. The Next Capability Offset field must be 000h or greater than 0FFh and the lower 2 bits of this field must be 00b.
11. The default values of each of the defined bits in the Uncorrectable Error Status register must be 0 with the exception of bit 0 (undefined) which is not checked.
(Note: The default value requirement applies to all devices except Root Complexes).
12. Each defined bit in the Uncorrectable Error Mask register must read zero by default with the exception of the following: bit 0 (undefined) which is not checked; bit 22 (Uncorrectable Internal Error) which is checked to default to 1 if the bit is writeable or default to 0 if it is not writeable. Test software writes 1 to each of these bits and verifies that 1 is read back. Test software records each bit that cannot be set to 1 (Unsupported uncorrectable error bits). If a required bit (bits 4, 12, 16, 18, 20 and for non-Switches bit 14) cannot be set to 1, then the test fails.
13. Test software writes all Uncorrectable Error Mask bits to their default values.
14. Test software checks the default values of each bit in the Uncorrectable Error Severity register defined bits with the exception of bit 0 (undefined) which is not checked. Test software then writes the opposite value to each bit and checks that the written value is returned.
Note: This rule does not apply to bits corresponding to Uncorrectable Error Mask bits that are not implemented as writeable.
(Note: The default value requirement applies to all devices except Root Complexes.)
15. Test software writes all Uncorrectable Error Severity bits to their default values.
16. Test software checks that all values for defined bits in the Correctable Error Status register are zero.
(Note: The default value requirement applies to all devices except Root Complexes.)
17. Test software checks that the Correctable Error Mask registers all read zero by default with the exception of the following: bit 13 (Advisory Non-Fatal Error) which is checked to default to 1; bit 14 (Corrected Internal Error) which is checked to default to 1 if the bit is writeable or default to 0 if it is not writeable; bit 15 (Header Log Overflow) which is checked to default to 1 if the bit is writeable or default to 0 if it is not writeable. Test software records each bit that cannot be set (Unsupported correctable error bits). If a required bit (bits 6, 7, 8, 12, 13) cannot be set, then the test fails.
18. Test software writes all Correctable Error Mask bits to their default values.
19. The Advanced Error Capabilities and Control register is examined. The ECRC Generation Enable and the ECRC Check Enable fields must default to zero. For Base 2.x or later testing: the Multiple Header Recording Enable fields must also default to zero.

- a. If the ECRC Check Capable bit is 1, test software checks to ensure that the ECRC Check Enable bit is writeable and can be set to 1 and cleared to 0.
- b. If the ECRC Generation Capable bit is 1, test software checks to ensure that the ECRC Generation Enable bit is writeable and can be set to 1 and cleared to 0.
- c. For Base 2.x or later testing; if the Multiple Header Recording Capable bit is 1, test software checks to ensure that the Multiple Header Recording Enable bit is writeable and can be set to 1 and cleared to 0.

(Note: The default value requirement applies to all devices except Root Complexes.)

20. For Root Ports and Root Complex Event Collectors only: test software checks that the Root Error Status register defaults to zero, excluding bits 31-27 (Advanced Error Interrupt Message Number) which are not checked.
21. For Root Ports and Root Complex Event Collectors only: test software checks that the Error Source Identification register defaults to zero.
22. For Bridges only: the default values of each of the defined bits in the Secondary Uncorrectable Error Status register must be zero.
23. For Bridges only: each defined bit in the Secondary Uncorrectable Error Mask register must read zero by default with the exception of the following: bit 3 (Received Master-Abort); bit 5 (Unexpected Split Completion Error); bit 7 (Uncorrectable Data Error Mask); bit 8 (Uncorrectable Attribute Error); bit 9 (Uncorrectable Address Error); bit 10 (Delayed Transaction Discard Timer Expired); bit 12 (SERR# Assertion Detected) all of which are checked to default to 1 if the bit is writeable or default to 0 if it is not writeable. Test software writes 1 to each of these bits and verifies that 1 is read back. Test software records each bit that cannot be set to 1 (Unsupported secondary uncorrectable error bits). If a required bit (bits 1, 3, 5-13) cannot be set to 1, then the test fails.
24. For Bridges only: test software writes all Secondary Uncorrectable Error Mask bits to their default values.
25. For Bridges only: test software checks the default values of each bit in the Secondary Uncorrectable Error Severity register defined bits. Test software then writes the opposite value to each bit and checks that the written value is returned.
Note: This rule does not apply to bits corresponding to Secondary Uncorrectable Error Mask bits that are not implemented as writeable.
26. For Bridges only: test software writes all Secondary Uncorrectable Error Severity bits to their default values.
27. The following register field characteristic checks are performed:

Advanced Error Reporting Extended Capability Header (Offset 00h) — DWORD

- | | |
|---------------------------------------|----|
| a. PCI Express Extended Capability ID | RO |
| b. Capability Version | RO |
| c. Next Capability Offset | RO |

Uncorrectable Error Status Register (Offset 04h) — DWORD

- | | |
|---|---------|
| a. RsvdZ_3-1 | RO-Zero |
| b. Data Link Protocol Error Status
(for VFs) | RO-Zero |
| (otherwise) | RW1CS |

c.	Surprise Down Error Status (Optional) (Downstream Ports with Surprise Down Error Capable as 1) (Downstream Ports with Surprise Down Error Capable as 0) (Upstream Ports or VFs)	RW1CS RO-Zero RO-Zero
d.	RsvdZ_11-6	RO-Zero
e.	Poisoned TLP Status	RW1CS
f.	Flow Control Protocol Error Status (Optional-Mask) (for VFs) (otherwise)	RO-Zero RW1CS
g.	Completion Timeout Status (Switches) (any device other than a Switch) Note: For a Switch Port there is no way to tell whether Completion Timeout Status needs to be RW1CS since there is no way to determine if the Switch may act as a requester on its own behalf.	RW1CS or RO-Zero RW1CS
h.	Completer Abort Status (Optional-Mask)	RW1CS
i.	Unexpected Completion Status	RW1CS
j.	Receiver Overflow Status (Optional-Mask) (for VFs) (otherwise)	RO-Zero RW1CS
k.	Malformed TLP Status (for VFs) (otherwise)	RO-Zero RW1CS
l.	ECRC Error Status (Optional) (for VFs) (non-VFs with ECRC Check Capable is 1) (non-VFs with ECRC Check Capable is 0)	RO-Zero RW1CS RO-Zero
m.	Unsupported Request Error Status	RW1CS
n.	ACS Violation Status (Optional) (Any function without ACS Extended Capability) (Downstream Ports with ACS Extended Capability) (Upstream Ports in a single function device) (Upstream Ports in multi-function device with ACS P2P Egress Control as 0) (Upstream Ports in multi-function device with ACS P2P Egress Control as 1)	RO-Zero RW1CS RO-Zero RO-Zero RW1CS
o.	Uncorrectable Internal Error Status (For Base 2.x or later testing: Optional-Mask)	RW1CS
p.	MC Blocked TLP Status (For Base 2.x or later testing: Optional) (Any function without Multicast Extended Capability) (Any function with Multicast Extended Capability)	RO-Zero RW1CS
q.	AtomicOp Egress Blocked Status (For Base 2.x or later testing: Optional) (If AtomicOp Routing Supported is 1) (If AtomicOp Routing Supported is 0)	RW1CS RO-Zero
r.	TLP Prefix Blocked Error Status	

- (For Base 2.x or later testing: Optional)
 (Root Ports and Switches, with End-End TLP Prefix Supported as 0) RO-Zero
 (Root Ports and Switches, with End-End TLP Prefix Supported as 1) RW1CS
 (any function other than Root Port or Switch) RO-Zero
- s. RsvdZ_31-26
 (For Base 2.x or later testing) RO-Zero
- t. RsvdZ_31-22
 (For Base 1.x testing) RO-Zero
- Note: These tests do not apply to those bits marked (Optional-Mask) if the corresponding mask is not implemented as writeable. In that case the bit is tested as RO-Zero.

Uncorrectable Error Mask Register (Offset 08h) — DWORD

- a. RsvdP_3-1 RO-Zero
- b. Data Link Protocol Mask
 (for VFs) RO-Zero
 (otherwise) RWS
- c. Surprise Down Error Mask (Optional)
 (Downstream Ports with Surprise Down Error Capable as 1) RWS
 (Downstream Ports with Surprise Down Error Capable as 0) RO-Zero
 (Upstream Ports or VFs) RO-Zero
- d. RsvdP_11-6 RO-Zero
- e. Poisoned TLP Mask
 (for VFs) RO-Zero
 (otherwise) RWS
- f. Flow Control Protocol Error Mask (Optional-Mask)
 (for VFs) RO-Zero
 (otherwise) RWS or RO-Zero
- g. Completion Timeout Mask
 (Switches) RWS or RO-Zero
 (for VFs) RO-Zero
 (any non-VF device other than a Switch) RWS
- Note: For a Switch Port there is no way to tell whether Completion Timeout Mask needs to be RWS since there is no way to determine if the Switch may act as a requester on its own behalf.
- h. Completer Abort Mask (Optional-Mask)
 (for VFs) RO-Zero
 (otherwise) RWS or RO-Zero
- i. Unexpected Completion Mask
 (for VFs) RO-Zero
 (otherwise) RWS
- j. Receiver Overflow Mask (Optional-Mask)
 (for VFs) RO-Zero
 (otherwise) RWS or RO-Zero

k.	Malformed TLP Mask (for VFs) (otherwise)	RO-Zero RWS
l.	ECRC Error Mask (Optional) (for VFs) (non-VFs with ECRC Check Capable is 1) (non-VFs with ECRC Check Capable is 0)	RO-Zero RWS RO-Zero
m.	Unsupported Request Error Mask (for VFs) (otherwise)	RO-Zero RWS
n.	ACS Violation Mask (Optional) (for VFs, or any function without ACS Extended Capability) (Downstream Ports with ACS Extended Capability) (Upstream Ports in a single function device) (Upstream Ports in multi-function device with ACS P2P Egress Control as 0) (non-VF Upstream Ports in multi-function device with ACS P2P Egress Control as 1)	RO-Zero RWS RO-Zero RO-Zero RWS
o.	Uncorrectable Internal Error Mask (For Base 2.x or later testing: Optional-Mask)	RWS or RO-Zero
p.	MC Blocked TLP Mask (For Base 2.x or later testing: Optional) (Any function without Multicast Extended Capability) (Any function with Multicast Extended Capability)	RO-Zero RWS
q.	AtomicOp Egress Blocked Mask (For Base 2.x or later testing: Optional) (If AtomicOp Routing Supported is 1) (If AtomicOp Routing Supported is 0)	RWS RO-Zero
r.	TLP Prefix Blocked Error Mask (For Base 2.x or later testing: Optional) (Root Ports and Switches, with End-End TLP Prefix Supported as 0) (Root Ports and Switches, with End-End TLP Prefix Supported as 1) (any function other than Root Port or Switch)	RO-Zero RWS RO-Zero
s.	RsvdP_31-26 (For Base 2.x or later testing)	RO-Zero
t.	RsvdP_31-22 (For Base 1.x testing)	RO-Zero

Note: Those bits that are marked (Optional-Mask) are checked to see if they are writeable or non-writeable. If one of those bits is writeable then that specific bit is considered implemented for the Status/Mask/Severity registers. Otherwise that specific bit is considered unimplemented for the Status/Mask/Severity registers. A writeable bit will be tested as RWS. A non-writeable bit will be tested as RO-Zero.

Uncorrectable Error Severity Register (Offset 0Ch) — DWORD

a. RsvdP_3-1	RO-Zero
b. Data Link Protocol Error Severity (for VFs) (otherwise)	RO-Zero RWS
c. Surprise Down Error Severity (Optional) (Downstream Ports with Surprise Down Error Capable as 1) (Downstream Ports with Surprise Down Error Capable as 0) (for VFs) (non-VF Upstream Ports)	RWS RO RO-Zero RO
d. RsvdP_11-6	RO-Zero
e. Poisoned TLP Severity (for VFs) (otherwise)	RO-Zero RWS
f. Flow Control Protocol Error Severity (Optional-Mask) (for VFs) (otherwise)	RO-Zero RWS
g. Completion Timeout Severity (Switches) (for VFs) (any non-VF device other than a Switch)	RWS or RO-Zero RO-Zero RWS
Note: For a Switch Port there is no way to tell whether Completion Timeout Severity needs to be RWS since there is no way to determine if the Switch may act as a requester on its own behalf.	
h. Completer Abort Error Severity (Optional-Mask) (for VFs) (otherwise)	RO-Zero RWS
i. Unexpected Completion Severity (for VFs) (otherwise)	RO-Zero RWS
j. Receiver Overflow Severity (Optional-Mask) (for VFs) (otherwise)	RO-Zero RWS
k. Malformed TLP Severity (for VFs) (otherwise)	RO-Zero RWS
l. ECRC Error Severity (Optional) (for VFs) (non-VFs with ECRC Check Capable is 1) (non-VFs with ECRC Check Capable is 0)	RO-Zero RWS RO
m. Unsupported Request Error Severity (for VFs) (otherwise)	RO-Zero RWS

n.	ACS Violation Severity (Optional)	RO-Zero
	(for VFs)	RO
	(for any non-VF without ACS Extended Capability)	RWS
	(Downstream Ports with ACS Extended Capability)	RO
	(Upstream Ports in a single function device)	RO
	(Upstream Ports in multi-function device with ACS P2P Egress Control as 0)	RO
	(Upstream Ports in multi-function device with ACS P2P Egress Control as 1)	RWS
o.	Uncorrectable Internal Error Severity	
	(For Base 2.x or later testing: Optional-Mask)	RWS
p.	MC Blocked TLP Severity	
	(For Base 2.x or later testing: Optional)	
	(Any function without Multicast Extended Capability)	RO
	(Any function with Multicast Extended Capability)	RWS
q.	AtomicOp Egress Blocked Severity	
	(For Base 2.x or later testing: Optional)	
	(If AtomicOp Routing Supported is 1)	RWS
	(If AtomicOp Routing Supported is 0)	RO
r.	TLP Prefix Blocked Error Severity	
	(For Base 2.x or later testing: Optional)	
	(Root Ports and Switches, with End-End TLP Prefix Supported as 0)	RO
	(Root Ports and Switches, with End-End TLP Prefix Supported as 1)	RWS
	(any function other than Root Port or Switch)	RO
s.	RsvdP_31-26	
	(For Base 2.x or later testing)	RO-Zero
t.	RsvdP_31-22	
	(For Base 1.x testing: bits 31-22)	RO-Zero

Note: These tests do not apply to those bits marked (Optional-Mask) if the corresponding mask is not implemented as writeable. In that case the bit must be RO (as default values for severity registers are implementation-specific).

Correctable Error Status Register (Offset 10h) — DWORD

a.	Receiver Error Status (Optional-Mask)	
	(for VFs)	RO-Zero
	(otherwise)	RW1CS
b.	RsvdZ_5-1	RO-Zero
c.	Bad TLP Status	
	(for VFs)	RO-Zero
	(otherwise)	RW1CS
d.	Bad DLLP Status	
	(for VFs)	RO-Zero
	(otherwise)	RW1CS
e.	REPLAY_NUM Rollover Status	
	(for VFs)	RO-Zero
	(otherwise)	RW1CS
f.	RsvdZ_11-9	RO-Zero

g. Replay Timer Timeout Status (for VFs) (otherwise)	RO-Zero RW1CS
h. Advisory Non-Fatal Error Status	RW1CS
i. Corrected Internal Error Status (For Base 2.x or later testing: Optional-Mask)	RW1CS
j. Header Log Overflow Status (For Base 2.x or later testing: Optional-Mask)	RW1CS
k. RsvdZ_31-16 (For Base 2.x or later testing)	RO-Zero
l. RsvdZ_31-14 (For Base 1.x testing)	RO-Zero

Note: These tests do not apply to those bits marked (Optional-Mask) if the corresponding mask is not implemented as writeable. In that case the bit must be RO-Zero.

Correctable Error Mask Register (Offset 14h) — DWORD

a. Receiver Error Mask (Optional-Mask) (for VFs) (otherwise)	RO-Zero RWS or RO-Zero RO-Zero
b. RsvdP_5-1	
c. Bad TLP Mask (for VFs) (otherwise)	RO-Zero RWS
d. Bad DLLP Mask (for VFs) (otherwise)	RO-Zero RWS
e. REPLAY_NUM Rollover Mask (for VFs) (otherwise)	RO-Zero RWS
f. RsvdP_11-9	RO-Zero
g. Replay Timer Timeout Mask (for VFs) (otherwise)	RO-Zero RWS
h. Advisory Non-Fatal Error Mask (for VFs) (otherwise)	RO-Zero RWS
i. Corrected Internal Error Mask (For Base 2.x or later testing: Optional-Mask)	RWS or RO-Zero
j. Header Log Overflow Mask (For Base 2.x or later testing: Optional-Mask)	RWS or RO-Zero

- | | | |
|----|--|---------|
| k. | RsvdP_31-16
(For Base 2.x or later testing) | RO-Zero |
| l. | RsvdP_31-14
(For Base 1.x testing) | RO-Zero |

Note: Those bits that are marked (Optional-Mask) are checked to see if they are writeable or non-writeable. If one of those bits is writeable then that specific bit is considered implemented for the Status/Mask registers. Otherwise that specific bit is considered unimplemented for the Status/Mask registers. A writeable bit will be tested as RWS. A non-writeable bit will be tested as RO-Zero.

Advanced Error Capabilities and Control Register (Offset 18h) — DWORD

- | | | |
|----|---|-------------------------------------|
| a. | First Error Pointer | ROS |
| b. | ECRC Generation Capable | RO |
| c. | ECRC Generation Enable (Optional)
(for VFs)
(non-VFs with ECRC Generation Capable is 1)
(non-VFs with ECRC Generation Capable is 0) | RO-Zero
RWS
RO-Zero or
RWS |
| d. | ECRC Check Capable | RO |
| e. | ECRC Check Enable (Optional)
(for VFs)
(non-VFs with ECRC Check Capable is 1)
(non-VFs with ECRC Check Capable is 0) | RO-Zero
RWS
RO-Zero or
RWS |
| f. | Multiple Header Recording Capable
(For Base 2.x or later testing) | RO |
| g. | Multiple Header Recording Enable
(For Base 2.x or later testing: Optional)
(if Multiple Header Recording Capable is 1)
(if Multiple Header Recording Capable is 0) | RWS
RO-Zero or
RWS |
| h. | TLP Prefix Log Present
(For Base 2.x or later testing: Optional)
(if End-End TLP Prefix Supported is 1)
(if End-End TLP Prefix Supported is 0) | ROS
RO-Zero |
| i. | RsvdP_31-12
(For Base 2.x or later testing) | RO-Zero |
| j. | RsvdP_31-9
(For Base 1.x testing) | RO-Zero |

Header Log Register (Offset 1Ch) — 4 DWORDS

- | | | |
|----|------------------------------|-----|
| a. | Header Log register (1st DW) | ROS |
| b. | Header Log register (2nd DW) | ROS |
| c. | Header Log register (3rd DW) | ROS |
| d. | Header Log register (4th DW) | ROS |

Root Error Command Register (Offset 2Ch) — DWORD

(Root Port and RC Event Collector)

- | | |
|---------------------------------------|---------|
| a. Correctable Error Reporting Enable | RW |
| b. Non-Fatal Error Reporting Enable | RW |
| c. Fatal Error Reporting Enable | RW |
| d. RsvdP_31-3 | RO-Zero |

Root Error Command Register (Offset 2Ch) — DWORD

(For Base 2.x or later testing)

(for all device types with AER Capability Version of 2h or greater and the End-End TLP Prefix Supported field as 1, except: Root Port, RC Event Collector, PCI Express to PCI/PCI-X Bridge, or PCI/PCI-X to PCI Express Bridge.)

- | | |
|---------------|---------|
| a. RsvdP_31-0 | RO-Zero |
|---------------|---------|

Root Error Status Register (Offset 30h) — DWORD

(Root Port and RC Event Collector)

- | | |
|--|---------|
| a. ERR_COR Received | RW1CS |
| b. Multiple ERR_COR Received | RW1CS |
| c. ERR_FATAL/NONFATAL Received | RW1CS |
| d. Multiple ERR_FATAL/NONFATAL Received | RW1CS |
| e. First Uncorrectable Fatal | RW1CS |
| f. Non-Fatal Error Messages Received | RW1CS |
| g. Fatal Error Messages Received | RW1CS |
| h. RsvdZ_26-7 | RO-Zero |
| i. Advanced Error Interrupt Message Number | RO |

Root Error Status Register (Offset 30h) — DWORD

(For Base 2.x or later testing)

(for all device types with AER Capability Version of 2h or greater and the End-End TLP Prefix Supported field as 1, except: Root Port, RC Event Collector, PCI Express to PCI/PCI-X Bridge, or PCI/PCI-X to PCI Express Bridge.)

- | | |
|---------------|---------|
| a. RsvdZ_31-0 | RO-Zero |
|---------------|---------|

Error Source Identification Register (Offset 34h) — DWORD

(Root Port and RC Event Collector)

- | | |
|---|-----|
| a. ERR_COR Source Identification | ROS |
| b. ERR_FATAL/NONFATAL Source Identification | ROS |

Error Source Identification Register (Offset 34h) — DWORD

(For Base 2.x or later testing)

(for all device types with AER Capability Version of 2h or greater and the End-End TLP Prefix Supported field as 1, except: Root Port, RC Event Collector, PCI Express to PCI/PCI-X Bridge, or PCI/PCI-X to PCI Express Bridge.)

- | | |
|---------------|---------|
| a. RsvdP_31-0 | RO-Zero |
|---------------|---------|

TLP Prefix Log Register (Offset 38h) — 4 DWORDS

(For Base 2.x or later testing)

(for all device types with AER Capability Version of 2h or greater and the End-End TLP Prefix Supported field as 1, except: PCI Express to PCI/PCI-X Bridge, or PCI/PCI-X to PCI Express Bridge.)

- | | |
|-------------------------------------|-----|
| a. TLP Prefix Log register (1st DW) | ROS |
| b. TLP Prefix Log register (2nd DW) | ROS |
| c. TLP Prefix Log register (3rd DW) | ROS |
| d. TLP Prefix Log register (4th DW) | ROS |

Secondary Uncorrectable Error Status Register (Offset 2Ch) — DWORD

(for PCI Express to PCI/PCI-X Bridge or PCI/PCI-X to PCI Express Bridge)

- | | |
|---|---------|
| a. Target-Abort on Split Completion Status (Optional-Mask) | RW1CS |
| b. Master-Abort on Split Completion Status | RW1CS |
| c. Received Target-Abort Status (Optional-Mask) | RW1CS |
| d. Received Master-Abort Status | RW1CS |
| e. RsvdZ_4 | RO-Zero |
| f. Unexpected Split Completion Error Status | RW1CS |
| g. Uncorrectable Split Completion Message Data Error Status | RW1CS |
| h. Uncorrectable Data Error Status | RW1CS |
| i. Uncorrectable Attribute Error Status | RW1CS |
| j. Uncorrectable Address Error Status | RW1CS |
| k. Delayed Transaction Discard Timer Expired Status | RW1CS |
| l. PERR# Assertion Detected | RW1CS |
| m. SERR# Assertion Detected | RW1CS |
| n. Internal Bridge Error Status | RW1CS |
| o. RsvdZ_31-14 | RO-Zero |

Note: These tests do not apply to those bits marked (Optional-Mask) if the corresponding mask is not implemented as writeable. In that case the bit must be RO-Zero.

Secondary Uncorrectable Error Mask Register (Offset 30h) — DWORD

(for PCI Express to PCI/PCI-X Bridge or PCI/PCI-X to PCI Express Bridge)

- | | |
|---|-------------------|
| a. Target-Abort on Split Completion Mask (Optional-Mask) | RWS or
RO-Zero |
| b. Master-Abort on Split Completion Mask | RWS |
| c. Received Target-Abort Mask (Optional-Mask) | RWS or
RO-Zero |
| d. Received Master-Abort Mask | RWS |
| e. RsvdP_4 | RO-Zero |
| f. Unexpected Split Completion Error Mask | RWS |
| g. Uncorrectable Split Completion Message Data Error Mask | RWS |
| h. Uncorrectable Data Error Mask | RWS |
| i. Uncorrectable Attribute Error Mask | RWS |
| j. Uncorrectable Address Error Mask | RWS |
| k. Delayed Transaction Discard Timer Expired Mask | RWS |
| l. PERR# Assertion Detected Mask | RWS |

m. SERR# Assertion Detected Mask	RWS
n. Internal Bridge Error Mask	RWS
o. RsvdP_31-14	RO-Zero

Note: Those bits that are marked (Optional-Mask) are checked to see if they are writeable or non-writeable. If one of those bits is writeable then that specific bit is considered implemented for the Status/Mask/Severity registers. Otherwise that specific bit is considered unimplemented for the Status/Mask/Severity registers. A writeable bit will be tested as RWS. A non-writeable bit will be tested as RO-Zero.

Secondary Uncorrectable Error Severity Register (Offset 34h) — DWORD

(for PCI Express to PCI/PCI-X Bridge or PCI/PCI-X to PCI Express Bridge)

a. Target-Abort on Split Completion Severity (Optional-Mask)	RWS
b. Master-Abort on Split Completion Severity	RWS
c. Received Target-Abort Severity (Optional-Mask)	RWS
d. Received Master-Abort Severity	RWS
e. RsvdP_4	RO-Zero
f. Unexpected Split Completion Error Severity	RWS
g. Uncorrectable Split Completion Message Data Error Severity	RWS
h. Uncorrectable Data Error Severity	RWS
i. Uncorrectable Attribute Error Severity	RWS
j. Uncorrectable Address Error Severity	RWS
k. Delayed Transaction Discard Timer Expired Severity	RWS
l. PERR# Assertion Detected Severity	RWS
m. SERR# Assertion Detected Severity	RWS
n. Internal Bridge Error Severity	RWS
o. RsvdP_31-14	RO-Zero

Note: These tests do not apply to those bits marked (Optional-Mask) if the corresponding mask is not implemented as writeable. In that case the bit must be RO-Zero.

Secondary Error Capabilities and Control Register (Offset 38h) — DWORD

(for PCI Express to PCI/PCI-X Bridge or PCI/PCI-X to PCI Express Bridge)

a. Secondary Uncorrectable First Error Pointer	ROS
b. RsvdP_31-5	RO-Zero

Secondary Header Log Register (Offset 3Ch) — 4 DWORDS

(for PCI Express to PCI/PCI-X Bridge or PCI/PCI-X to PCI Express Bridge)

a. Transaction Attribute	ROS
b. Transaction Command Lower	ROS
c. Transaction Command Upper	ROS
d. RsvdZ_63-44	RO-Zero
e. Transaction Address	ROS

28. The following default value checks are performed:

Uncorrectable Error Status Register Default Value (Offset 04h) — DWORD

a. Data Link Protocol Error Status (for non-VFs)	0
b. Surprise Down Error Status (Optional) (Downstream Ports with Surprise Down Error Capable as 1)	0
c. Poisoned TLP Status	0

d. Flow Control Protocol Error Status (Optional-Mask) (for non-VFs)	0
e. Completion Timeout Status (any device other than a Switch)	0
f. Completer Abort Status (Optional-Mask)	0
g. Unexpected Completion Status	0
h. Receiver Overflow Status (Optional-Mask) (for non-VFs)	0
i. Malformed TLP Status (for non-VFs)	0
j. ECRC Error Status (Optional) (non-VFs with ECRC Check Capable is 1)	0
k. Unsupported Request Error Status	0
l. ACS Violation Status (Optional) (Downstream Ports with ACS Extended Capability) (Upstream Ports in multi-function device with ACS P2P Egress Control as 1)	0
m. Uncorrectable Internal Error Status (For Base 2.x or later testing: Optional-Mask)	0
n. MC Blocked TLP Status (For Base 2.x or later testing: Optional) (Any function with Multicast Extended Capability)	0
o. AtomicOp Egress Blocked Status (For Base 2.x or later testing: Optional) (If AtomicOp Routing Supported is 1)	0
p. TLP Prefix Blocked Error Status (For Base 2.x or later testing: Optional) (Root Ports and Switches, with End-End TLP Prefix Supported as 1)	0

Note: These tests do not apply to those bits marked (Optional-Mask) if the corresponding mask is not implemented as writeable. In that case the default value is not checked.

Uncorrectable Error Mask Register Default Value (Offset 08h) — DWORD

a. Data Link Protocol Mask (for non-VFs)	0
b. Surprise Down Error Mask (Optional) (Downstream Ports with Surprise Down Error Capable as 1)	0
c. Poisoned TLP Mask (for non-VFs)	0
d. Flow Control Protocol Error Mask (Optional-Mask) (for non-VFs)	0
e. Completion Timeout Mask (any non-VF device other than a Switch)	0
f. Completer Abort Mask (Optional-Mask) (for non-VFs)	0
g. Unexpected Completion Mask (for non-VFs)	0
h. Receiver Overflow Mask (Optional-Mask) (for non-VFs)	0

i. Malformed TLP Mask (for non-VFs)	0
j. ECRC Error Mask (Optional) (non-VFs with ECRC Check Capable is 1)	0
k. Unsupported Request Error Mask (for non-VFs)	0
l. ACS Violation Mask (Optional) (Downstream Ports with ACS Extended Capability) (non-VF Upstream Ports in multi-function device with ACS P2P Egress Control as 1)	0
m. Uncorrectable Internal Error Mask (For Base 2.x or later testing: Optional-Mask)	1
n. MC Blocked TLP Mask (For Base 2.x or later testing: Optional) (Any function with Multicast Extended Capability)	0
o. AtomicOp Egress Blocked Mask (For Base 2.x or later testing: Optional) (If AtomicOp Routing Supported is 1)	0
p. TLP Prefix Blocked Error Mask (For Base 2.x or later testing: Optional) (Root Ports and Switches, with End-End TLP Prefix Supported as 1)	0

Note: These tests do not apply to those bits that are marked (Optional-Mask) if the bit is not implemented as writeable. In that case the default value is not checked.

Uncorrectable Error Severity Register Default Value (Offset 0Ch) — DWORD

a. Data Link Protocol Error Severity (for non-VFs)	1
b. Surprise Down Error Severity (Optional) (Downstream Ports with Surprise Down Error Capable as 1)	1
c. Poisoned TLP Severity (for non-VFs)	0
d. Flow Control Protocol Error Severity (Optional-Mask) (for non-VFs)	1
e. Completion Timeout Severity (any non-VF device other than a Switch)	0
f. Completer Abort Error Severity (Optional-Mask) (for non-VFs)	0
g. Unexpected Completion Severity (for non-VFs)	0
h. Receiver Overflow Severity (Optional-Mask) (for non-VFs)	1
i. Malformed TLP Severity (for non-VFs)	1
j. ECRC Error Severity (Optional) (non-VFs with ECRC Check Capable is 1)	0
k. Unsupported Request Error Severity (for non-VFs)	0

l.	ACS Violation Severity (Optional) (Downstream Ports with ACS Extended Capability) (Upstream Ports in multi-function device with ACS P2P Egress Control as 1)	0
m.	Uncorrectable Internal Error Severity (Optional-Mask) (For Base 2.x or later testing)	1
n.	MC Blocked TLP Severity (Optional) (For Base 2.x or later testing) (Any function with Multicast Extended Capability)	0
o.	AtomicOp Egress Blocked Severity (Optional) (For Base 2.x or later testing) (If AtomicOp Routing Supported is 1)	0
p.	TLP Prefix Blocked Error Severity (Optional) (For Base 2.x or later testing) (Root Ports and Switches, with End-End TLP Prefix Supported as 1)	0

Note: These tests do not apply to those bits that are marked (Optional-Mask) if the corresponding mask is not implemented as writeable. In that case the default value is not checked.

Correctable Error Status Register Default Value (Offset 10h) — DWORD

a.	Receiver Error Status (Optional-Mask) (for non-VFs)	0
b.	Bad TLP Status (for non-VFs)	0
c.	Bad DLLP Status (for non-VFs)	0
d.	REPLAY_NUM Rollover Status (for non-VFs)	0
e.	Replay Timer Timeout Status (for non-VFs)	0
f.	Advisory Non-Fatal Error Status	0
g.	Corrected Internal Error Status (For Base 2.x or later testing: Optional-Mask)	0
h.	Header Log Overflow Status (For Base 2.x or later testing: Optional-Mask)	0

Note: These tests do not apply to those bits marked (Optional-Mask) if the corresponding mask is not implemented as writeable. In that case the default value is not checked.

Correctable Error Mask Register Default Value (Offset 14h) — DWORD

a.	Receiver Error Mask (Optional-Mask) (for non-VFs)	0
b.	Bad TLP Mask (for non-VFs)	0
c.	Bad DLLP Mask (for non-VFs)	0
d.	REPLAY_NUM Rollover Mask (for non-VFs)	0

e. Replay Timer Timeout Mask (for non-VFs)	0
f. Advisory Non-Fatal Error Mask (for non-VFs)	1
g. Corrected Internal Error Mask (For Base 2.x or later testing: Optional-Mask)	1
h. Header Log Overflow Mask (For Base 2.x or later testing: Optional-Mask)	1

Note: These tests do not apply to those bits that are marked (Optional-Mask) if the bit is not implemented as writeable. In that case the default value is not checked.

Advanced Error Capabilities and Control Register Default Value (Offset 18h) — DWORD

a. ECRC Generation Enable (Optional) (non-VFs with ECRC Generation Capable is 1)	0
b. ECRC Check Enable (Optional) (non-VFs with ECRC Check Capable is 1)	0
c. Multiple Header Recording Enable (For Base 2.x or later testing: Optional) (if Multiple Header Recording Capable is 1)	0
d. TLP Prefix Log Present (For Base 2.x or later testing: Optional) (if End-End TLP Prefix Supported is 1)	0

Root Error Command Register Default Value (Offset 2Ch) — DWORD

(Root Port and RC Event Collector)

a. Correctable Error Reporting Enable	0
b. Non-Fatal Error Reporting Enable	0
c. Fatal Error Reporting Enable	0

Root Error Status Register Default Value (Offset 30h) — DWORD

(Root Port and RC Event Collector)

a. ERR_COR Received	0
b. Multiple ERR_COR Received	0
c. ERR_FATAL/NONFATAL Received	0
d. Multiple ERR_FATAL/NONFATAL Received	0
e. First Uncorrectable Fatal	0
f. Non-Fatal Error Messages Received	0
g. Fatal Error Messages Received	0

Error Source Identification Register Default Value (Offset 34h) — DWORD

(Root Port and RC Event Collector)

a. ERR_COR Source Identification	0000h
b. ERR_FATAL/NONFATAL Source Identification	0000h

Secondary Uncorrectable Error Status Register Default Value (Offset 2Ch) — DWORD

(for PCI Express to PCI/PCI-X Bridge or PCI/PCI-X to PCI Express Bridge)

a. Target-Abort on Split Completion Status (Optional-Mask)	0
b. Master-Abort on Split Completion Status	0
c. Received Target-Abort Status (Optional-Mask)	0

d. Received Master-Abort Status	0
e. Unexpected Split Completion Error Status	0
f. Uncorrectable Split Completion Message Data Error Status	0
g. Uncorrectable Data Error Status	0
h. Uncorrectable Attribute Error Status	0
i. Uncorrectable Address Error Status	0
j. Delayed Transaction Discard Timer Expired Status	0
k. PERR# Assertion Detected	0
l. SERR# Assertion Detected	0
m. Internal Bridge Error Status	0

Note: These tests do not apply to those bits marked (Optional-Mask) if the corresponding mask is not implemented as writeable. In that case the default value is not checked.

Secondary Uncorrectable Error Mask Register Default Value (Offset 30h) — DWORD

(for PCI Express to PCI/PCI-X Bridge or PCI/PCI-X to PCI Express Bridge)

a. Target-Abort on Split Completion Mask (Optional-Mask)	0
b. Master-Abort on Split Completion Mask	0
c. Received Target-Abort Mask (Optional-Mask)	0
d. Received Master-Abort Mask	1
e. Unexpected Split Completion Error Mask	1
f. Uncorrectable Split Completion Message Data Error Mask	0
g. Uncorrectable Data Error Mask	1
h. Uncorrectable Attribute Error Mask	1
i. Uncorrectable Address Error Mask	1
j. Delayed Transaction Discard Timer Expired Mask	1
k. PERR# Assertion Detected Mask	0
l. SERR# Assertion Detected Mask	1
m. Internal Bridge Error Mask	0

Note: These tests do not apply to those bits that are marked (Optional-Mask) if the bit is not implemented as writeable. In that case the default value is not checked.

Secondary Uncorrectable Error Severity Register Default Value (Offset 34h) — DWORD

(for PCI Express to PCI/PCI-X Bridge or PCI/PCI-X to PCI Express Bridge)

a. Target-Abort on Split Completion Severity (Optional-Mask)	0
b. Master-Abort on Split Completion Severity	0
c. Received Target-Abort Severity (Optional-Mask)	0
d. Received Master-Abort Severity	0
e. Unexpected Split Completion Error Severity	0
f. Uncorrectable Split Completion Message Data Error Severity	1
g. Uncorrectable Data Error Severity	0
h. Uncorrectable Attribute Error Severity	1
i. Uncorrectable Address Error Severity	1
j. Delayed Transaction Discard Timer Expired Severity	0
k. PERR# Assertion Detected Severity	0

l. SERR# Assertion Detected Severity	1
m. Internal Bridge Error Severity	0

Note: These tests do not apply to those bits marked (Optional-Mask) if the corresponding mask is not implemented as writeable. In that case the default value is not checked.

29. For functions under test that have a link, the test is run at each of the following link speeds:
 - a. 2.5 GT/s (for devices that are capable). See Section 2.1.2.12 for details.
 - b. 5.0 GT/s (for devices that are capable). See Section 2.1.2.13 for details.
 - c. 8.0 GT/s (for devices that are capable). See Section 2.1.2.14 for details.
30. If the device type is Root Port or Root Complex Integrated Endpoint, then the test is repeated using each RCRB associated with the function under test, or associated with an RCRB that is itself associated with the function under test. See Section 2.1.2.17 for details. All offsets in the tests are now relative to the RCRB's base address and all cycles are now memory space accesses.

The test *fails* if:

- ☐ More than one Advanced Error Reporting capability structure is present.
- ☐ An RCRB contains an Advanced Error Reporting capability structure.
- ☐ A Root Port or a Root Complex Event Collector does not implement the Root Error Command, Root Error Status, or Error Source Identification registers.
- ☐ The Capability Version field does not report 2h or 1h (if End-End TLP Prefixes not supported).
- ☐ The Next Capability Offset field does not report 000h or a value greater than 0FFh or the lower two bits are not 00b.
- ☐ Any of the defined bits in the Uncorrectable Error Status register report an error (after standard configuration performed by the test).
- ☐ A required bit in the Uncorrectable Error Mask register cannot be set.
- ☐ A required bit in the Uncorrectable Error Severity register cannot be set.
- ☐ Any of the defined bits in the Correctable Error Status register report an error (after standard configuration performed by the test).
- ☐ A required bit in the Correctable Error Mask register cannot be set.
- ☐ ECRC Check Capable is 1 and the ECRC Check Enable field is not writeable.
- ☐ ECRC Generation Capable is 1 and the ECRC Generation Enable field is not writeable.
- ☐ Multiple Header Recording Capable bit is 1 and the Multiple Header Recording Enable field is not writeable (for Base 2.x or later testing only).
- ☐ A Root Port or a Root Complex Event Collector has the Root Error Status register report an error.
- ☐ A Root Port or a Root Complex Event Collector has the Error Source Identification register report an error.
- ☐ A Bridge has any of the defined bits in the Secondary Uncorrectable Error Status register report an error (after standard configuration performed by the test).

- ☐ A Bridge has a required bit in the Secondary Uncorrectable Error Mask register that cannot be set.
- ☐ A Bridge has a required bit in the Secondary Uncorrectable Error Severity register that cannot be set.
- ☐ Any of the register field characteristic tests fail.
- ☐ Any of the default value tests fail.

2.2.12. TD_1_8 Virtual Channel Extended Capability Structure

The test verifies that if the function under test reports a PCI Express Virtual Channel Capability structure, it is implemented as defined in the relevant specifications.

Relevant Specifications

- ☐ *PCI Express Base Specification*
- ☐ *Single Root I/O Virtualization and Sharing Specification*

(See Section 1.3 for additional specification revision details.)

Applicable Device/Port Types

This test is run on all device/port types.

This test is run on any RCRB associated with the function under test.

Starting Configuration

This test is run with functions starting in the D0-Uninitialized state following the standard initialization procedures in Section 2.1.1. Testing in additional Dx states may be optionally supported. ASPM states of the upstream and downstream ports are disabled for this test.

Overview of Test Steps

Test software performs the following steps:

1. Configure the function under test following the procedure described in Section 2.1.1.
2. Examine the Extended Capability ID field for each of the detected Extended Capabilities. Determine how many instances of either the Extended Capability ID of 0002h (Virtual Channel Extended Capability) or the Extended Capability ID of 0009h (Virtual Channel Extended Capability) are found. If more than one is found, the test terminates with a failure.
3. If the Extended Capability ID of 0002h is found in a VF, the test terminates with a failure.
4. If the Extended Capability ID of 0009h is found in an RCRB, the test terminates with a failure.
5. If the Extended Capability ID of 0009h is found in a VF, the test terminates with a failure.
6. If an Extended Capability ID of 0002h or if an Extended Capability ID of 0009h is found the following checks are performed on that extended capability structure:
7. The Capability Version field must be 1h.
8. The Next Capability Offset field must be 000h or greater than 0FFh and the lower 2 bits of this field must be 00b.

9. Test software reads the DWORD at offset 04h (Port VC Capability Register 1) and performs the following checks:
 - a. The Low Priority Extended VC Count field is less than or equal to the Extended VC Count field value.
 - b. The Reference Clock field must be 00b.
10. Test software reads the DWORD from offset 08h (Port VC Capability Register 2 register) and performs the following checks:
 - a. No reserved bits (4-7) in the VC Arbitration Capability field are set.
 - b. The VC Arbitration Capability field must not be zero if the Low Priority Extended VC Count field value is non-zero.
 - c. The VC Arbitration Capability field must be zero if the Low Priority Extended VC Count field value is zero.
 - d. The VC Arbitration Table Offset field must not be zero if any of bits 1, 2, or 3 of the VC Arbitration Capability field are set.
11. Test software reads the DWORD from offset 0Ch (Port VC Control register and Port VC Status register) and performs the following checks:
 - a. The VC Arbitration Table Status field returns zero.
12. If the default VC Arbitration Select field value is a WRR based arbitration method test software reads the default values in the VC Arbitration table. These values must all be zero.
13. Test software writes each of the supported data values indicated by the VC Arbitration Capability field value to the VC Arbitration Select field and verifies the value is read back.
14. For each supported WRR arbitration field mechanism, test software tests loading the table by writing a table of all zeroes of the appropriate size to the indicated VC Arbitration table offset. Test software then checks that the VC Arbitration Table Status field returns 1. Test software then writes 1 to the Load VC Arbitration Table field. Test software then reads the VC Arbitration Table Status field repeatedly until it returns 0. If it does not return 0 within 1 second, the test fails.
15. Step 14 is repeated using all VC entries for the highest supported VC instead of zero.
16. Test software verifies that VC Resource Control, VC Resource Status, and VC Resource Capability registers are present for each VC supported by the function (as indicated by the Extended VC Count field value).
17. For each set of VC Resource Control, VC Resource Status, and VC Resource Capability registers as reported in the Extended VC Count field value, test software performs the following checks:
 - a. The Port Arbitration Capability field: For a Switch Port or RCRB it must not have bit 6 or 7 set (reserved).
 - b. The Port Arbitration Table Offset field: For a Switch Port or RCRB it must be nonzero if the Port Arbitration Capability field value indicates any WRR based methods.
 - c. The TC/VC Map field must be FFh for the first VC resource (VC0) by default. For all other VCs it must be 00h by default. Test software writes each valid TC combination to this field and ensures that the same value can be read back, with the exception of bit 0 which is RO=1 for the first VC resource (VC0) and RO=0 for all other VCs.
(Note: This default value restriction does not apply to an RCRB.)
 - d. The Load Port Arbitration Table field: For a Switch Port or RCRB it must be zero.

- e. The Port Arbitration Select field: For a Switch Port or RCRB – Test software writes each supported data value (based on the Port Arbitration Capability field value) and ensures that the same value is read back.
 - f. The VC-ID field: Test software ensures that the value is hard-wired to 000b for the first VC resource (VC0).
 - g. The VC Enable field: Test software checks that this field is hardwired to one for the first VC resource (VC0).
 - h. The Port Arbitration Table Status field: For a Switch Port or RCRB it must be zero.
18. For each supported WRR based Port Arbitration scheme test software writes a table with all values of the minimum port number. After the first write to the table, test software checks that the Port Arbitration Table Status field returns 1. Test software writes 1 to the Load Port Arbitration field. Test software then reads the Port Arbitration Table Status field repeatedly until it returns 0. If it does not return 0 within 1 second, the test fails.
19. Step 18 is repeated with a valid arbitration table containing all ports.
20. The following register field characteristic checks are performed:

Virtual Channel Extended Capability Header (Offset 00h) — DWORD

- a. PCI Express Extended Capability ID RO
- b. Capability Version RO
- c. Next Capability Offset RO

Port VC Capability Register 1 (Offset 04h) — DWORD

- a. Extended VC Count RO
- b. RsvdP_3 RO-Zero
- c. Low Priority Extended VC Count RO
- d. RsvdP_7 RO-Zero
- e. Reference Clock RO
- f. Port Arbitration Table Entry Size
(for Endpoints) RO-Zero
(for non-Endpoints) RO
- g. RsvdP_31-12 RO-Zero

Port VC Capability Register 2 (Offset 08h) — DWORD

- a. VC Arbitration Capability RO
- b. RsvdP_23-8 RO-Zero
- c. VC Arbitration Table Offset RO

Port VC Control Register (Offset 0Ch) — WORD

- a. Load VC Arbitration Table RO-Zero
- b. VC Arbitration Select
(if more than one VC in the LPVC group is not enabled) RW
(if more than one VC in the LPVC group is enabled) RO
(only those values in VC Arbitration Capability are written)
- c. RsvdP_15-4 RO-Zero

Port VC Status Register (Offset 0Eh) — WORD

- a. VC Arbitration Table Status RO
- b. RsvdZ_15-1 RO-Zero

21. For each VC Resource register set value [n] (given by value in Extended VC Count), the following register field characteristic checks and register field default value checks are performed:

VC Resource Capability Register (n) (Offset 10h + n * 0Ch) — DWORD

- | | |
|---|---------|
| a. Port Arbitration Capability | RO |
| b. RsvdP_13-8 | RO-Zero |
| c. Undefined_14 | RO |
| Note: This was once Advanced Packet Switching. | |
| d. Reject Snoop Transactions
(for non-FLR testing) | HwInit |
| (for FLR testing) | RO |
| e. Maximum Time Slots
(for non-FLR testing) | HwInit |
| (for FLR testing) | RO |
| f. RsvdP_23 | RO-Zero |
| g. Port Arbitration Table Offset | RO |

VC Resource Control Register (n) (Offset 14h + n * 0Ch) — DWORD

- | | |
|---|---------|
| a. TC/VC Map
(for VC0: bit 0) | RO-Ones |
| (for non-VC0: bit 0) | RO-Zero |
| (bits 7-1) | RW |
| b. RsvdP_15-8 | RO-Zero |
| c. Load Port Arbitration Table | RO-Zero |
| d. Port Arbitration Select
(only those values reported in Port Arbitration Capability are written) | RW |
| e. RsvdP_23-20 | RO-Zero |
| f. VC ID
(for VC0) | RO-Zero |
| (for all non VC0) | RW |
| g. RsvdP_30-27 | RO-Zero |
| h. VC Enable
(for VC0) | RO |
| (for all non-VC0) | RW |

Reserved Register (n) (Offset 18h + n * 0Ch) — WORD

- | | |
|---------------|---------|
| a. RsvdP_15-0 | RO-Zero |
|---------------|---------|

VC Resource Status Register (n) (Offset 1Ah + n * 0Ch) — WORD

- | | |
|----------------------------------|---------|
| a. Port Arbitration Table Status | RO |
| b. VC Negotiation Pending | RO |
| c. RsvdZ_15-2 | RO-Zero |

VC Resource Control Register (n) Default Value (Offset 14h + n * 0Ch) – DWORD

- | | |
|-------------------|-----|
| a. TC/VC Map | |
| (for VC0) | FFh |
| (for all non VC0) | 00h |
| b. VC Enable | |
| (for VC0) | 1 |
| (for all non-VC0) | 0 |
22. For functions under test that have a link, the test is run at each of the following link speeds:
- 2.5 GT/s (for devices that are capable). See Section 2.1.2.12 for details.
 - 5.0 GT/s (for devices that are capable). See Section 2.1.2.13 for details.
 - 8.0 GT/s (for devices that are capable). See Section 2.1.2.14 for details.
23. If the device type is Root Port or Root Complex Integrated Endpoint, then the test is repeated using each RCRB associated with the function under test, or associated with an RCRB that is itself associated with the function under test. See Section 2.1.2.17 for details. All offsets in the tests are now relative to the RCRB's base address and all cycles are now memory space accesses.

The test *fails* if:

- ☐ More than one Virtual Channel capability structure is present.
- ☐ A VF contains a Virtual Channel capability structure with Extended Capability ID of 0002h or 0009h.
- ☐ An RCRB contains a Virtual Channel capability structure with Extended Capability ID of 0009h.
- ☐ The Capability Version field does not report 1h.
- ☐ The Next Capability Offset field does not report 000h or a value greater than 0FFh or the lower two bits are not 00b.
- ☐ The Low Priority Extended VC Count field is greater than the Extended VC Count field (Port VC Capability Register 1).
- ☐ The Reference Clock field is not zero (Port VC Capability Register 1).
- ☐ The VC Arbitration Capability field contains a reserved value.
- ☐ The VC Arbitration Capability field contains a zero value and the Low Priority Extended VC Count field is non-zero.
- ☐ The VC Arbitration Capability field contains a non-zero value and the Low Priority Extended VC count field is zero.
- ☐ The VC Arbitration Table Offset field is zero when WRR based VC Arbitration Capabilities are supported.
- ☐ The VC Arbitration Table Status field reads non-zero by default.
- ☐ The default VC Arbitration Select field value is a WRR based arbitration method and the default VC Arbitration table values are not all zero.
- ☐ VC Resource Control, VC Resource Status, and VC Resource Capability registers are not present for each VC supported by the function (as indicated by Extended VC Count field).
- ☐ A valid Port Arbitration Select field value or Port Arbitration table cannot be written.

- ☐ Any of the described VC Resource Control and VC Resource Status register value checks are not met.
- ☐ Any of the register field characteristic tests fail.
- ☐ Any of the default value tests fail.

2.2.13. TD_1_9 Device Serial Number Extended Capability Structure

The test verifies that if the function under test reports a PCI Express Device Serial Number Capability structure, it is implemented as defined in the relevant specifications.

Relevant Specifications

- ☐ *PCI Express Base Specification*
- ☐ *Single Root I/O Virtualization and Sharing Specification*

(See Section 1.3 for additional specification revision details.)

Applicable Device/Port Types

This test is run on all device/port types.

This test is run on any RCRB associated with the function under test.

Starting Configuration

This test is run with functions starting in the D0-Uninitialized state following the standard initialization procedures in Section 2.1.1. Testing in additional Dx states may be optionally supported. ASPM states of the upstream and downstream ports are disabled for this test.

Overview of Test Steps

Test software performs the following steps:

1. Configure the function under test following the procedure described in Section 2.1.1.
2. Examine the Extended Capability ID field for each of the detected Extended Capabilities. Determine how many instances of the Extended Capability ID of 0003h (Device Serial Number Extended Capability) are found. If more than one is found, the test terminates with a failure.
3. If the Extended Capability ID of 0003h is found in a VF, the test terminates with a failure.
4. If the Extended Capability ID of 0003h is found in an RCRB, the test terminates with a failure.
5. If an Extended Capability ID of 0003h is found for an extended capability the following checks are performed on that extended capability structure:
 - a. The Capability Version field must be 1h.
 - b. The Next Capability Offset field must be 000h or greater than 0FFh and the lower 2 bits of this field must be 00b.
6. The DWORD at offset 04h (Serial Number register Lower DW) is read. The test logs a warning if the value is zero.
7. The DWORD at offset 08h (Serial Number register Upper DW) is read. The test logs a warning if the value is zero.
8. Both Serial Number registers are checked to ensure they are read only.

9. The following register field characteristic checks are performed:

Device Serial Number Extended Capability Header (Offset 00h) — DWORD

- | | |
|---------------------------------------|----|
| a. PCI Express Extended Capability ID | RO |
| b. Capability Version | RO |
| c. Next Capability Offset | RO |

Serial Number Register (Offset 04h) — 2 DWORDS

- | | |
|--------------------------------------|----|
| a. Serial Number register (Lower DW) | RO |
| b. Serial Number register (Upper DW) | RO |

10. For functions under test that have a link, the test is run at each of the following link speeds:
- 2.5 GT/s (for devices that are capable). See Section 2.1.2.12 for details.
 - 5.0 GT/s (for devices that are capable). See Section 2.1.2.13 for details.
 - 8.0 GT/s (for devices that are capable). See Section 2.1.2.14 for details.
11. If the device type is Root Port or Root Complex Integrated Endpoint, then the test is repeated using each RCRB associated with the function under test, or associated with an RCRB that is itself associated with the function under test. See Section 2.1.2.17 for details. All offsets in the tests are now relative to the RCRB's base address and all cycles are now memory space accesses.

The test *fails* if:

- ☐ More than one Device Serial Number capability structure is present.
- ☐ A VF contains a Device Serial Number capability structure.
- ☐ An RCRB contains a Device Serial Number capability structure.
- ☐ The Capability Version field does not report 1h.
- ☐ The Next Capability Offset field does not report 000h or a value greater than 0FFh or the lower two bits are not 00b.
- ☐ Either of the Serial Number registers read zero.
- ☐ Either of the Serial Number registers is not read only.
- ☐ Any of the register field characteristic tests fail.

2.2.14. TD_1_10 Power Budgeting Extended Capability Structure

The test verifies that if the function under test reports a PCI Express Power Budgeting Capability structure, it is implemented as defined in the relevant specifications.

Relevant Specifications

- ☐ *PCI Express Base Specification*
- ☐ *Single Root I/O Virtualization and Sharing Specification*

(See Section 1.3 for additional specification revision details.)

Applicable Device/Port Types

This test is run on all device/port types.

This test is run on any RCRB associated with the function under test.

Starting Configuration

This test is run with functions starting in the D0-Uninitialized state following the standard initialization procedures in Section 2.1.1. Testing in additional Dx states may be optionally supported. ASPM states of the upstream and downstream ports are disabled for this test.

Overview of Test Steps

Test software performs the following steps:

1. Configure the function under test following the procedure described in Section 2.1.1.
2. Examine the Extended Capability ID field for each of the detected Extended Capabilities. Determine how many instances of the Extended Capability ID of 0004h (Power Budgeting Extended Capability) are found. If more than one is found, the test terminates with a failure.
3. If the Extended Capability ID of 0004h is found in a VF, the test terminates with a failure.
4. If the Extended Capability ID of 0004h is found in an RCRB, the test terminates with a failure.
5. If an Extended Capability ID of 0004h is found for an extended capability the following checks are performed on that extended capability structure:
 - a. The Capability Version field must be 1h.
 - b. The Next Capability Offset field must be 000h or greater than 0FFh and the lower 2 bits of this field must be 00b.
6. Test software sets the Data Select Value [DSV] to 00h.
7. Test software writes [DSV] to the Data Select register.
8. Test software reads a DWORD from offset 8h (Data register).
9. If the returned Data register value is non-zero, test software performs the following checks on the Data register:
 - a. Test software checks that the Power Rail field is one of the defined values (000b, 001b, 010b, or 111b).
 - b. Test software checks that the Type field of the Power Budgeting data is one of the defined values (000b, 001b, 010b, 011b, or 111b).
 - c. Test software records the PM State and Type fields for Power Budgeting data.
 - d. If the Data Scale field is 00b then the Base Power field must not be F3h to FFh.
10. Test software increments [DSV] by 1 and repeats steps 7-10. Test software continues repeating until the Data register returns 0000 0000h, or until [DSV] exceeds FFh. The value written to the Data Select register for the instance when the Data register first returns 0000 0000h determines the out of bound index value [OBI] which is recorded. If the Data register does not return 0000 0000h before [DSV] exceeds FFh, then [OBI] is set to 100h.
11. If [DSV] does not yet exceed FFh, test software increments [DSV] by 1 and repeats steps 7, 8, 11, but now checks that the Data register only returns 0000 0000h. If the Data register does not return 0000 0000h, a failure is recorded, but the test continues. Test software continues repeating until [DSV] exceeds FFh.

12. Test software checks that at least one of the recorded Power Budget data values contains the following information for each value of Power Rail that was recorded from the device:
 - a. The PM State field is equal to 00b (D0) and the Type field is equal to 111b (Maximum).
 - b. The PM State field is equal to 00b (D0) and the Type field is equal to 011b (Sustained).
13. Test software repeats steps 7-12 with the [DSV] value written to the Data Select register in inverse order (starting with [OBI]-1 and decrementing to 00h). The returned information must be the same.
14. If the out of bound index value [OBI] is equal to or less than FFh, test software repeats steps 7-12 with the [DSV] value written to the Data Select register starting with [OBI] for the first access, and then using 00h for the second access, incrementing which each access up to [OBI]-1. The returned information must be the same.
15. The following register field characteristic checks are performed:

Power Budgeting Extended Capability Header (Offset 00h) — DWORD

- | | |
|---------------------------------------|----|
| a. PCI Express Extended Capability ID | RO |
| b. Capability Version | RO |
| c. Next Capability Offset | RO |

Data Select Register (Offset 04h) — BYTE

- | | |
|----------------|----|
| a. Data Select | RW |
|----------------|----|

Reserved Register (Offset 05h) — 3 BYTES

- | | |
|---------------|---------|
| a. RsvdP_23-0 | RO-Zero |
|---------------|---------|

Data (n) Register (Offset 08h) — DWORD

Data Select register is written with value (n) prior to this register test. This test is then repeated for all values of (n) = 00h to ([OBI]-1) where [OBI] is the first Data Select value that returns all zeroes in the Data register.

- | | |
|-----------------|---------|
| a. Base Power | RO |
| b. Data Scale | RO |
| c. PM Sub State | RO |
| d. PM State | RO |
| e. Type | RO |
| f. Power Rail | RO |
| g. RsvdP_31-21 | RO-Zero |

Note: Since the Data Select register is not sticky, it must be re-written following any Reset Sequence (described in Section 2.1.1.1) being executed while testing this register.

Data (n) Register (Offset 08h) — DWORD

If [OBI] is less than or equal to FFh, then Data Select register is written with value (n) prior to this register test. This test is then repeated for all values of (n) = [OBI] to FFh where [OBI] is the first Data Select value that returns all zeroes in the Data register.

- | | |
|---------------|---------|
| a. RsvdP_31-0 | RO-Zero |
|---------------|---------|

Note: Since the Data Select register is not sticky, it must be re-written following any Reset Sequence (described in Section 2.1.1.1) being executed while testing this register.

Power Budget Capability Register (Offset 0Ch) — BYTE

- | | |
|--|---------|
| a. System Allocated
(for non-FLR testing) | HwInit |
| (for FLR testing) | RO |
| b. RsvdP_7-1 | RO-Zero |

Reserved Register (Offset 0Dh) — 3 BYTES

- | | |
|---------------|---------|
| a. RsvdP_23-0 | RO-Zero |
|---------------|---------|

16. For functions under test that have a link, the test is run at each of the following link speeds:
 - a. 2.5 GT/s (for devices that are capable). See Section 2.1.2.12 for details.
 - b. 5.0 GT/s (for devices that are capable). See Section 2.1.2.13 for details.
 - c. 8.0 GT/s (for devices that are capable). See Section 2.1.2.14 for details.
17. If the device type is Root Port or Root Complex Integrated Endpoint, then the test is repeated using each RCRB associated with the function under test, or associated with an RCRB that is itself associated with the function under test. See Section 2.1.2.17 for details. All offsets in the tests are now relative to the RCRB's base address and all cycles are now memory space accesses.

The test *fails* if:

- ☐ More than one Power Budgeting capability structure is present.
- ☐ A VF contains a Power Budgeting capability structure.
- ☐ An RCRB contains a Power Budgeting capability structure.
- ☐ The Capability Version field does not report 1h.
- ☐ The Next Capability Offset field does not report 000h or a value greater than 0FFh or the lower two bits are not 00b.
- ☐ The function under test reports an invalid Power Rail field value for any of the non-zero Power Budgeting data sets that it reports.
- ☐ The function under test reports an invalid Type field value for any of the non-zero Power Budgeting data sets that it reports.
- ☐ The device does not report at least a D0 Max and D0 Sustained set of power data for every power rail it uses power from.
- ☐ The device reports more than one set of data for the same PM State, Type, and Power Rail combination.
- ☐ The device reports a non-zero set of data for a Data Select register value greater than the number of supported data sets.
- ☐ Any of the register field characteristic tests fail.
- ☐ The power information is not returned normally when valid indices are used after an out of bound index is written to the Data Select register.

2.2.15. TD_1_11 Command and Status Registers

The test verifies that the function under test implements the Command and Status registers as defined in the relevant specifications.

Relevant Specifications

- ❑ *PCI Express Base Specification*
 - ❑ *PCI Local Bus Specification, Revision 3.0*
 - ❑ *Single Root I/O Virtualization and Sharing Specification*
- (See Section 1.3 for additional specification revision details.)

Applicable Device/Port Types

This test is run on all device/port types.

Starting Configuration

This test is run with functions starting in the D0-Uninitialized state following the standard initialization procedures in Section 2.1.1. Testing in additional Dx states may be optionally supported. ASPM states of the upstream and downstream ports are disabled for this test.

Overview of Test Steps

Test software performs the following steps:

1. Configure the function under test following the procedure described in Section 2.1.1.
2. The following register field characteristic checks are performed:

Command Register (Location 04h) — WORD

- | | |
|--|------------------|
| a. I/O Space Enable | |
| (for VFs) | RO-Zero |
| (otherwise) | RW or
RO-Zero |
| b. Memory Space Enable | |
| (for VFs) | RO-Zero |
| (any function with a Type 0 Configuration Space Header excluding VFs) | RW or
RO-Zero |
| (any function with a Type 1 Configuration Space Header) | RW |
| c. Bus Master Enable | |
| (any function with a Type 0 Configuration Space Header) | RW or
RO-Zero |
| (any function with a Type 1 Configuration Space Header) | RW |
| d. Special Cycle Enable | RO-Zero |
| e. Memory Write and Invalidate | |
| (for PCI Express to PCI/PCI-X Bridges
or PCI/PCI-X to PCI Express Bridges) | RW or
RO-Zero |
| (all except for PCI Express to PCI/PCI-X Bridge
or PCI/PCI-X to PCI Express Bridge) | RO-Zero |
| f. VGA Palette Snoop | |
| (for PCI/PCI-X to PCI Express Bridges) | RW or
RO-Zero |

(all except for PCI/PCI-X to PCI Express Bridge)	RO-Zero
g. Parity Error Response (for VFs) (for RC Integrated Endpoints)	RO-Zero RW or RO-Zero
(any function excluding VFs and RC Integrated Endpoints)	RW
h. IDSEL Sleeping/Wait Cycle Control	RO-Zero
i. SERR# Enable (for VFs) (for RC Integrated Endpoints)	RO-Zero RW or RO-Zero
(any function excluding VFs and RC Integrated Endpoints)	RW
j. Fast Back-to-Back Transactions Enable (for PCI/PCI-X to PCI Express Bridges)	RW or RO-Zero
(all except for PCI/PCI-X to PCI Express Bridge)	RO-Zero
k. Interrupt Disable (for VFs) (for non-VFs where Interrupt Pin is non-zero) (for non-VFs where Interrupt Pin is zero)	RO-Zero RW RW or RO-Zero
l. Reserved_15-11	RO-Zero

Status Register (Location 06h) — WORD

a. Reserved_2-0	RO-Zero
b. Interrupt Status (for VFs) (otherwise)	RO-Zero RO
c. Capabilities List	RO-Ones
d. 66 MHz Capable (for PCI/PCI-X to PCI Express Bridges) (all except for PCI/PCI-X to PCI Express Bridge)	RO RO-Zero
e. Reserved_6	RO-Zero
f. Fast Back-to-Back Transactions Capable (for PCI/PCI-X to PCI Express Bridges) (all except for PCI/PCI-X to PCI Express Bridge)	RO RO-Zero
g. Master Data Parity Error	RW1C
h. DEVSEL Timing (for PCI/PCI-X to PCI Express Bridges) (all except for PCI/PCI-X to PCI Express Bridge)	RO RO-Zero
i. Signaled Target Abort	RW1C
j. Received Target Abort	RW1C
k. Received Master Abort	RW1C
l. Signaled System Error	RW1C
m. Detected Parity Error	RW1C

3. The following default value checks are performed:

Command Register Default Value (Location 04h) — WORD

a. I/O Space Enable	0
b. Memory Space Enable	0
c. Bus Master Enable	0
d. Memory Write and Invalidate	0
e. VGA Palette Snoop	0
f. Parity Error Response	0
g. SERR# Enable	0
h. Fast Back-to-Back Transactions Enable	0
i. Interrupt Disable	0

Status Register Default Value (Location 06h) — WORD

a. Interrupt Status	0
b. Master Data Parity Error	0
c. Signaled Target Abort	0
d. Received Target Abort	0
e. Received Master Abort	0
f. Signaled System Error	0
g. Detected Parity Error	0

Note: The zero default values on the Status register assume that no events that would cause an interrupt to be generated, or an error to be detected, have occurred in the test environment.

4. For functions under test that have a link, the test is run at each of the following link speeds:
 - a. 2.5 GT/s (for devices that are capable). See Section 2.1.2.12 for details.
 - b. 5.0 GT/s (for devices that are capable). See Section 2.1.2.13 for details.
 - c. 8.0 GT/s (for devices that are capable). See Section 2.1.2.14 for details.

The test *fails* if:

- ☐ Any of the register field characteristic tests fail.
- ☐ Any of the default value tests fail.

2.2.16. TD_1_12 Cache Line Size, Master/Primary Latency Timer, and Min_Gnt/Max_Lat Registers

The test verifies that the function under test implements the Cache Line Size, Master/Primary Latency Timer, Min_Gnt, and Max_Lat register fields as defined in the relevant specifications.

Relevant Specifications

- ☐ *PCI Express Base Specification*
- ☐ *PCI Express to PCI/PCI-X Bridge Specification, Revision 1.0*
- ☐ *Single Root I/O Virtualization and Sharing Specification*

(See Section 1.3 for additional specification revision details.)

Applicable Device/Port Types

This test is run on all device/port types. (Min_Gnt and Max_Lat registers are only checked in devices that have a Type 0 configuration space header.)

Starting Configuration

This test is run with functions starting in the D0-Uninitialized state following the standard initialization procedures in Section 2.1.1. Testing in additional Dx states may be optionally supported. ASPM states of the upstream and downstream ports are disabled for this test.

Overview of Test Steps

Test software performs the following steps:

1. Configure the function under test following the procedure described in Section 2.1.1.
2. For a PCI/PCI-X to PCI Express Bridge the following checks are performed:
 - a. Write a byte of FFh to location 0Dh (Master/Primary Latency Timer register).
 - b. Read a byte from location 0Dh (Master/Primary Latency Timer register) and verify that one of the following values is returned: (FFh, F8h, or 10h to 00h).
 - c. Write a byte of 00h to location 0Dh (Master/Primary Latency Timer register).
 - d. Read a byte from location 0Dh (Master/Primary Latency Timer register) and verify that one of the following values is returned: (10h to 00h).
 - e. If the value returned is 00h, then record this by setting value [MLTNZ] to 0. If the value returned is non-zero, set value [MLTNZ] to 1. (Value [MLTNZ] will be used by the default value test to indicate that the Master/Primary Latency Timer register is not hard-wired to a non-zero value.
3. The following register field characteristic checks are performed:

Cache Line Size Register (Location 0Ch) — BYTE

(for VFs)

RO-Zero

(for PCI Express to PCI/PCI-X Bridges
or PCI/PCI-X to PCI Express Bridges)

RW or

RO-Zero

(all except for PCI Express to PCI/PCI-X Bridge
or PCI/PCI-X to PCI Express Bridge)

RW

Master/Primary Latency Timer Register (location 0Dh) — BYTE

(for PCI/PCI-X to PCI Express Bridges: bits 2-0)

RW or RO

(for PCI/PCI-X to PCI Express Bridges: bits 7-3)

RW or RO

(all except for PCI/PCI-X to PCI Express Bridge)

RO-Zero

Min_Gnt (Location 3Eh) — BYTE

(only if device has Type 0 configuration space header)

RO-Zero

Max_Lat (Location 3Fh) — BYTE

(only if device has Type 0 configuration space header)

RO-Zero

4. The following default value checks are performed:

Cache Line Size Register Default Value (Location 0Ch) — BYTE

0

Master/Primary Latency Timer Register Default Value (Location 0Dh) — BYTE

(for PCI/PCI-X to PCI Express Bridges)

(if [MLTNZ] is 0)

0

5. For functions under test that have a link, the test is run at each of the following link speeds:
 - a. 2.5 GT/s (for devices that are capable). See Section 2.1.2.12 for details.
 - b. 5.0 GT/s (for devices that are capable). See Section 2.1.2.13 for details.
 - c. 8.0 GT/s (for devices that are capable). See Section 2.1.2.14 for details.

The test *fails* if:

- ☐ For a PCI/PCI-X to PCI Express Bridge, the Master/Primary Latency Timer register is not implemented as RW, RW with lower 3 bits hard-wired to 000b, or RO with a value between 16 and 0 inclusive.
- ☐ Any of the register field characteristic tests fail.
- ☐ Any of the default value tests fail.

2.2.17. TD_1_13 Interrupt Pin and Interrupt Line Registers

The test verifies that the function under test implements the Interrupt Pin and Interrupt Line registers as defined in the relevant specifications. (If legacy interrupt support is indicated the function must implement either a MSI or a MSI-X capability structure, or both. The MSI and MSI-X capability structures' contents are tested elsewhere.)

Relevant Specifications

- ☐ *PCI Express Base Specification*
- ☐ *Single Root I/O Virtualization and Sharing Specification*

(See Section 1.3 for additional specification revision details.)

Applicable Device/Port Types

This test is run on all device/port types.

Starting Configuration

This test is run with functions starting in the D0-Uninitialized state following the standard initialization procedures in Section 2.1.1. Testing in additional Dx states may be optionally supported. ASPM states of the upstream and downstream ports are disabled for this test.

Overview of Test Steps

Test software performs the following steps:

1. Configure the function under test following the procedure described in Section 2.1.1.
2. Read the byte from location 3Dh (Interrupt Pin register) in configuration space and perform the following checks:
 - a. For a VF this value must be 00h.
 - b. For a non-VF this value must be one of the following: (00h, 01h, 02h, 03h, or 04h).
3. If the Interrupt Pin register has a valid non-zero value perform the following checks:
 - a. Examine the Capability ID field for each of the function's Capabilities. Determine how many instances of the Capability ID of 05h (MSI Capability) are found.
 - b. Examine the Capability ID field for each of the function's Capabilities. Determine how many instances of the Capability ID of 11h (MSI-X Capability) are found.
 - c. Verify that at least one instance of either the MSI capability structure or the MSI-X capability structure is implemented.

4. The following register field characteristic checks are performed:

Interrupt Line Register (Location 3Ch) — BYTE

(for VFs)
(otherwise)

RO-Zero
RW

Interrupt Pin Register (Location 3Dh) — BYTE

(for VFs)
(otherwise)

RO-Zero
RO

5. For functions under test that have a link, the test is run at each of the following link speeds:
- 2.5 GT/s (for devices that are capable). See Section 2.1.2.12 for details.
 - 5.0 GT/s (for devices that are capable). See Section 2.1.2.13 for details.
 - 8.0 GT/s (for devices that are capable). See Section 2.1.2.14 for details.

The test *fails* if:

- ☐ For a VF, the Interrupt Pin register does not return 00h.
- ☐ For a non-VF, the Interrupt Pin register does not return 00h, 01h, 02h, 03h, or 04h.
- ☐ The Interrupt Pin register is non-zero and neither a MSI capability structure nor a MSI-X capability structure are implemented.
- ☐ Any of the register field characteristic tests fail.

2.2.18. TD_1_14 Secondary Latency Timer and Secondary Status Registers

The test verifies that the function under test implements the Secondary Latency Timer and Secondary Status registers as defined in the relevant specifications.

Relevant Specifications

- ☐ *PCI Express Base Specification*
- ☐ *PCI Express to PCI/PCI-X Bridge Specification, Revision 1.0*

(See Section 1.3 for additional specification revision details.)

Applicable Device/Port Types

This test is run on all device/port types implementing Type 1 Configuration Space Headers.

Starting Configuration

This test is run with functions starting in the D0-Uninitialized state following the standard initialization procedures in Section 2.1.1. Testing in additional Dx states may be optionally supported. ASPM states of the upstream and downstream ports are disabled for this test.

Overview of Test Steps

Test software performs the following steps:

- Configure the function under test following the procedure described in Section 2.1.1.
- Read the Header Type field and only if bits 6-0 return 000 0001b (Type 1 Configuration Space Header) perform the following tests.

3. For a PCI Express to PCI/PCI-X Bridge the following checks are performed:
 - a. Write a byte of FFh to location 1Bh (Secondary Latency Timer register).
 - b. Read a byte from location 1Bh (Secondary Latency Timer register) and verify that one of the following values is returned: (FFh, F8h, or 10h to 00h).
 - c. Write a byte of 00h to location 1Bh (Secondary Latency Timer register).
 - d. Read a byte from location 1Bh (Secondary Latency Timer register) and verify that one of the following values is returned: (10h to 00h).
 - e. If the value returned is 00h, then record this by setting value [SLTNZ] to 0. If the value returned is non-zero, set value [SLTNZ] to 1. (Value [SLTNZ] will be used by the default value test to indicate that the Secondary Latency Timer register is not hard-wired to a non-zero value.)

4. The following register field characteristic checks are performed:

Secondary Latency Timer Register (Location 1Bh) — BYTE

(for PCI Express to PCI/PCI-X Bridges: bits 2-0) RW or RO

(for PCI Express to PCI/PCI-X Bridges: bits 7-3) RW or RO

(all except for PCI Express to PCI/PCI-X Bridge) RO-Zero

Secondary Status Register (Location 1Eh) — WORD

a. RsvdZ_4-0 RO-Zero

b. 66 MHz Capable
(all except for PCI Express to PCI/PCI-X Bridge) RO-Zero
(for PCI Express to PCI/PCI-X Bridges) RO

c. RsvdZ_6 RO-Zero

d. Fast Back-to-Back Transactions Capable
(all except for PCI Express to PCI/PCI-X Bridge) RO-Zero
(for PCI Express to PCI/PCI-X Bridges) RO

e. Master Data Parity Error RW1C

f. DEVSEL Timing
(all except for PCI Express to PCI/PCI-X Bridge) RO-Zero
(for PCI Express to PCI/PCI-X Bridges) RO

g. Signaled Target Abort RW1C

h. Received Target Abort RW1C

i. Received Master Abort RW1C

j. Received System Error RW1C

k. Detected Parity Error RW1C

5. The following default value checks are performed:

Secondary Latency Timer Register Default Value (Location 1Bh) — BYTE

(for PCI Express to PCI/PCI-X Bridges)

(if [SLTNZ] is 0) 0

Secondary Status Register Default Value (Location 1Eh) — WORD

a. Master Data Parity Error	0
b. Signaled Target Abort	0
c. Received Target Abort	0
d. Received Master Abort	0
e. Received System Error	0
f. Detected Parity Error	0

Note: The zero default values on the Secondary Status register assume that no events that would cause an error to be detected have occurred in the test environment.

6. For functions under test that have a link, the test is run at each of the following link speeds:
 - a. 2.5 GT/s (for devices that are capable). See Section 2.1.2.12 for details.
 - b. 5.0 GT/s (for devices that are capable). See Section 2.1.2.13 for details.
 - c. 8.0 GT/s (for devices that are capable). See Section 2.1.2.14 for details.

The test *fails* if:

- ☐ For a PCI Express to PCI/PCI-X Bridge, the Secondary Latency Timer register is not implemented as RW, RW with lower 3 bits hard-wired to 000b, or RO with a value between 16 and 0 inclusive.
- ☐ Any of the register field characteristic tests fail.
- ☐ Any of the default value tests fail.

2.2.19. TD_1_15 Bridge Control Register

The test verifies that the function under test implements the Bridge Control register as defined in the relevant specifications.

Relevant Specifications

- ☐ *PCI Express Base Specification*
- ☐ *PCI Express to PCI/PCI-X Bridge Specification, Revision 1.0*

(See Section 1.3 for additional specification revision details.)

Applicable Device/Port Types

This test is run on all device/port types implementing Type 1 configuration space headers.

Starting Configuration

This test is run with functions starting in the D0-Uninitialized state following the standard initialization procedures in Section 2.1.1. Testing in additional Dx states may be optionally supported. ASPM states of the upstream and downstream ports are disabled for this test.

Overview of Test Steps

Test software performs the following steps:

1. Configure the function under test following the procedure described in Section 2.1.1.
2. The following register field characteristic checks are performed:

Bridge Control Register (Location 3Eh) — WORD

a. Parity Error Response Enable	RW
b. SERR# Enable	RW
c. ISA Enable	RW
d. VGA Enable	RW or RO-Zero
e. VGA 16-bit Decode	RW or RO-Zero
f. Master Abort Mode (all except for PCI Express to PCI/PCI-X Bridge or PCI/PCI-X to PCI Express Bridge) (for PCI Express to PCI/PCI-X Bridges or PCI/PCI-X to PCI Express Bridges)	RO-Zero RW
g. Secondary Bus Reset	RW
h. Fast Back-to-Back Transactions Enable (all except for PCI Express to PCI/PCI-X Bridge) (for PCI Express to PCI/PCI-X Bridges)	RO-Zero RW or RO-Zero
i. Primary Discard Timer (all except for PCI/PCI-X to PCI Express Bridge) (for PCI/PCI-X to PCI Express Bridges)	RO-Zero RW
j. Secondary Discard Timer (all except for PCI Express to PCI/PCI-X Bridge) (for PCI Express to PCI/PCI-X Bridges)	RO-Zero RW
k. Discard Timer Status (all except for PCI Express to PCI/PCI-X Bridge or PCI/PCI-X to PCI Express Bridge) (for PCI Express to PCI/PCI-X Bridges or PCI/PCI-X to PCI Express Bridges)	RO-Zero RW1C
l. Discard Timer SERR# Enable (all except for PCI Express to PCI/PCI-X Bridge or PCI/PCI-X to PCI Express Bridge) (for PCI Express to PCI/PCI-X Bridges or PCI/PCI-X to PCI Express Bridges)	RO-Zero RW
m. RsvdP_15-12	RO-Zero

3. The following default value checks are performed:

Bridge Control Register Default Value (Location 3Eh) — WORD

- | | |
|--|---|
| a. Parity Error Response Enable | 0 |
| b. SERR# Enable | 0 |
| c. ISA Enable | 0 |
| d. VGA Enable | 0 |
| e. VGA 16-bit Decode | 0 |
| f. Master Abort Mode | 0 |
| g. Secondary Bus Reset | 0 |
| h. Fast Back-to-Back Transactions Enable | 0 |
| i. Primary Discard Timer | 0 |
| j. Secondary Discard Timer | 0 |
| k. Discard Timer Status | 0 |
| l. Discard Timer SERR# Enable | 0 |
4. For functions under test that have a link, the test is run at each of the following link speeds:
- a. 2.5 GT/s (for devices that are capable). See Section 2.1.2.12 for details.
 - b. 5.0 GT/s (for devices that are capable). See Section 2.1.2.13 for details.
 - c. 8.0 GT/s (for devices that are capable). See Section 2.1.2.14 for details.

The test *fails* if:

- ☐ Any of the register field characteristic tests fail.
- ☐ Any of the default value tests fail.

2.2.20. TD_1_16 PCI Power Management Capability Structure

The test verifies that the function under test reports a Power Management Capability structure and that it is implemented as defined in the relevant specifications.

Relevant Specifications

- ☐ *PCI Express Base Specification*
- ☐ *PCI Bus Power Management Interface Specification, Revision 1.2*
- ☐ *Single Root I/O Virtualization and Sharing Specification*

(See Section 1.3 for additional specification revision details.)

Applicable Device/Port Types

This test is run on all device/port types.

Starting Configuration

This test is run with functions starting in the D0-Uninitialized state following the standard initialization procedures in Section 2.1.1. Testing in additional Dx states may be optionally supported. ASPM states of the upstream and downstream ports are disabled for this test.

Overview of Test Steps

Test software performs the following steps:

1. Configure the function under test following the procedure described in Section 2.1.1.
2. Examine the Capability ID field for each of the function's Capabilities. Determine how many instances of the Capability ID of 01h (PCI Power Management Capability) are found. If more than one is found, the test terminates with a failure.
3. For all non-VFs, if the Capability ID of 01h is not found for a capability, the test terminates with a failure.
4. If a Capability ID of 01h is found for a capability, the following checks are performed.
5. Read the DWORD from offset 00h (Power Management Capabilities register) in the Power Management capability.
6. The Next Capability Pointer field must be 00h or greater than 3Fh and the lower 2 bits of this field must be 00b.
7. The Version field must be set to 011b.
8. The PME Clock field must be zero.
9. If device type is Root Port, or Switch Upstream Port, or Switch Downstream Port then PME Support field bits 31, 30, and 27 must each return 1.
10. For the WORD at offset 04h (Power Management Status and Control register) the following tests are performed:
 - a. Read a WORD from offset 04h and use this as the base PM value [PMV].
 - b. Clear to 00b the lower two bits of [PMV] and use this as the new [PMV] value (bits 15-2 contain what was read, and bits 1-0 are 00b).
 - c. Write a WORD to offset 04h using the data as [PMV] ORed with 0000h (Power State field = 00b).
 - d. Wait 10 seconds.
 - e. Read a WORD from offset 04h - the Power State field must return 00b.
 - f. If the D1 Support field is 1 (in Power Management Capabilities register), then write a WORD to offset 04h using the data as [PMV] ORed with 0001h (Power State field = 01b). Otherwise skip to step i).
 - g. Wait 10 seconds.
 - h. Read a WORD from offset 04h - the Power State field must return 01b.
 - i. If the D2 Support field is 1 (in Power Management Capabilities register), then write a WORD to offset 04h using the data as [PMV] ORed with 0002h (Power State field = 10b). Otherwise skip to step l).
 - j. Wait 10 seconds.
 - k. Read a WORD from offset 04h - the Power State field must return 10b.
 - l. Write a WORD to offset 04h using the data as [PMV] ORed with 0003h (Power State field = 11b).
 - m. Wait 10 seconds.
 - n. Read a WORD from offset 04h - the Power State field must return 11b.
 - o. Write a WORD to offset 04h using the data as [PMV] ORed with 0000h (Power State field = 00b).
 - p. Wait 10 seconds.

- q. Read a WORD from offset 04h - the Power State field must return 00b.
11. Test software sets the Data Present Flag [DPF] to 0.
 12. Test software writes a data value of Fh to the Data Select field and then reads back the value from the Data Select field and performs the following checks:
 - a. For a VF, the value must be 0h.
 - b. For a VF, skip to step 19.
 - c. For a non-VF, if the value is 0h, skip to step 19.
 13. Test software sets the Data Select Value [DSV] to 0h.
 14. Test software writes [DSV] to the Data Select field
 15. Test software reads back a byte from offset 07h (Data register) in the Power Management Capability and performs the following checks:
 - a. If the read byte is non-zero, then set [DPF] to 1.
 - b. Read the Data Scale field and if it is non-zero, then set [DPF] to 1.
 - c. If the data value written to the Data Select field is 0h to 7h, then if the read byte was non-zero, then the Data Scale field must be non-zero.
 - d. If the data value written to the Data Select field is 8h and the device is a multi function device, and the function under test is function 0, then if the read byte was non-zero then the Data Scale field must be non-zero.
 - e. If the data value written to Data Select field is 0h, 3h, 4h, 7h, then if the read byte was 00h a warning message is issued, but this is not treated in itself as a failure.
 - f. If the data value written to Data Select field is 1h or 5h and the D1 Support field returns 1, then if the read byte was 00h a warning message is issued, but this is not treated in itself as a failure.
 - g. If the data value written to Data Select field is 2h or 6h and the D2 Support field returns 1, then if the read byte was 00h a warning message is issued, but this is not treated in itself as a failure.
 - h. If the data value written to Data Select field is 8h and the device is a multi function device, and the function under test is function 0, then if the read byte was 00h a warning message is issued, but this is not treated in itself as a failure.
 - i. If the data value written to the Data Select field is 8h and the device is a single function device, or if the function under test is not function 0, then the read byte must be 00h and the Data Scale field must be 00b.
 - j. If the data value written to the Data Select field is 9h to Fh, then the read byte must be 00h and the Data Scale field must be 00b.
 - k. Write a new data value to the Data Scale field which is the inverse of the current read back value. Read back the Data Scale field again and ensure that it returns the original value (it is a RO field).
 - l. Write a new data value to the Data register which is the inverse of the current read back value. Read back the Data register again and ensure that it returns the original value (it is a RO field).
 16. Test software increments [DSV] by 1 and repeats steps 14-16. Test software continues repeating until [DSV] exceeds Fh.

17. Test software repeats steps 14-15, 17 with the [DSV] value written to the Data Select register in inverse order and stopping when it decrements below 0h (starting with Fh and decrementing to 0h). The returned information must be the same.
18. If [DPF] is 1, test software reads back a word from offset 02h (Power Management Capabilities register) in the Power Management Capability and performs the following checks:
 - a. Test software writes a 1 to the Auxiliary (AUX) Power PM Enable (Device Control register) and then reads it back, and only if it returns 0:
 - i. The Aux Current field must return 000b.
19. The following register field characteristic checks are performed:

Power Management Capability Header (Offset 00h) — WORD

- | | |
|----------------------------|----|
| a. Capability ID | RO |
| b. Next Capability Pointer | RO |

Power Management Capabilities Register (Offset 02h) — WORD

- | | |
|--|---------|
| a. Version | RO |
| b. PME Clock | RO-Zero |
| c. RsvdP_4 | RO-Zero |
| d. Device Specific Initialization | RO |
| e. Aux Current | |
| (For Root Ports, Switch Ports, and Bridges) | RO |
| (For non-Root Ports, non-Switch Ports, and non-Bridges, | |
| if Auxiliary (AUX) Power PM Enable (Device Control) is not RO-Zero or | |
| PME Support field is 1 xxxxb) | RO |
| (For non-Root Ports, non-Switch Ports, and non-Bridges, | |
| if Auxiliary (AUX) Power PM Enable (Device Control) is RO-Zero and | |
| PME Support field is 0 xxxxb) | RO-Zero |
| Note: For a Root Port or a Switch Port there is no way to tell whether Aux Current needs | |
| to be RO-Zero since the PME Support field must be 1 1xx1b for forwarding PME even if | |
| the device cannot generate PME. | |
| f. D1 Support | RO |
| g. D2 Support | RO |
| h. PME Support | RO |

Power Management Status/Control (n) Register (Offset 04h) — DWORD

Data Select register is written with value (n) prior to this register test. This test is then repeated for (n) = 0h to Fh.

- | | |
|---|---------|
| a. Power State | RW |
| b. RsvdP_2 | RO-Zero |
| c. No_Soft_Reset | RO |
| d. RsvdP_7-4 | RO-Zero |
| e. PME_En | |
| (If PME Support field is 0 0000b) | RW or |
| | RO-Zero |
| (For Root Ports, Switch Ports, and Bridges) | RWS or |
| | RW |

(For non-Root Ports, non-Switch Ports, and non-Bridges, if PME Support field is 1 xxxxb)		RWS
(For non-Root Ports, non-Switch Ports, and non-Bridges, if PME Support field is 0 xxxxb and non-zero)		RW
Note: For a Root Port or a Switch Port there is no way to tell whether PME_En needs to be RWS since the PME Support field must be 1 1xx1b for forwarding PME even if the device cannot generate PME.		
f. Data Scale (n)		
(for VFs)		RO-Zero
(otherwise)		RO
g. PME_Status		
(For Root Ports, Switch Ports, and Bridges)		RW1CS or RW1C
(For non-Root Ports, non-Switch Ports, and non-Bridges, if PME Support field is 1 xxxxb)		RW1CS
(For non-Root Ports, non-Switch Ports, and non-Bridges, if PME Support field is 0 xxxxb)		RW1C
Note: For a Root Port or a Switch Port there is no way to tell whether PME_Status needs to be RW1CS since the PME Support field must be 1 1xx1b for forwarding PME even if the device cannot generate PME.		
h. RsvdP_21-16		RO-Zero
i. B2/B3 Support		
(all except PCI Express to PCI/PCI-X Bridge)		RO-Zero
(for PCI Express to PCI/PCI-X Bridges)		RO
j. Bus Power/Clock Control Enable		
(all except PCI Express to PCI/PCI-X Bridge)		RO-Zero
(for PCI Express to PCI/PCI-X Bridges)		RO
k. Data (n)		
(For Base 1.x or Base 2.x testing: for VFs)		RO
(For Base 3.x or later testing: for VFs)		RO-Zero
(otherwise)		RO
Note: Since the Data Select register is not sticky, it must be re-written following any Reset Sequence (described in Section 2.1.1.1) being executed while testing this register.		
6. The following default value checks are performed:		

Power Management Status/Control Register Default Value (Offset 04h) — DWORD

- | | |
|--|---|
| a. Power State | 0 |
| b. PME_En
(For non-Root Ports, non-Switch Ports, and non-Bridges,
if PME Support field is 0 xxxxb) | 0 |
| c. PME_Status
(For non-Root Ports, non-Switch Ports, and non-Bridges,
if PME Support field is 0 xxxxb) | 0 |
7. For functions under test that have a link, the test is run at each of the following link speeds:
- 2.5 GT/s (for devices that are capable). See Section 2.1.2.12 for details.
 - 5.0 GT/s (for devices that are capable). See Section 2.1.2.13 for details.
 - 8.0 GT/s (for devices that are capable). See Section 2.1.2.14 for details.

The test *fails* if:

- ☐ For a non-VF, a PCI Power Management capability structure is not present.
- ☐ More than one Power Management capability structure is present.
- ☐ A non-zero Next Capability Pointer field is less than or equal to 3Fh, or the lower 2 bits are non-zero.
- ☐ The version for the Power Management Capability structure is not 011b.
- ☐ The PME Clock field is non-zero.
- ☐ A Root Port or a Switch Port does not have bits 31, 30, and 27 each return 1 in the PME Support field.
- ☐ The Power State field is not zero in the Power Management Status and Control register.
- ☐ The PME Support is non-zero and the PME_En field is not writeable.
- ☐ The Power State field cannot be set to D0, D1 (if supported), D2 (if supported), and D3.
- ☐ The Power State field does not read D0 after a reset when the function is in D0-Uninitialized or D0-Initialized state.
- ☐ For a VF, the Data Select register returns a non-zero value.
- ☐ The Data register contains unsupported entries.
- ☐ The Data Scale field contains unsupported entries.
- ☐ The Aux Current field is not zero when any Data register entry is non-zero and Auxiliary (AUX) Power PM Enable is RO-Zero.
- ☐ Any of the register field characteristic tests fail.
- ☐ Any of the default value tests fail.

2.2.21. TD_1_17 MSI-X Capability Structure

This test verifies that if the function under test reports a MSI-X capability structure, it is implemented as defined in the relevant specifications. (If legacy interrupt support is indicated the function must implement at least either a MSI or a MSI-X capability structure. The MSI capability structure is tested elsewhere.)

Relevant Specifications

- ❑ *PCI Express Base Specification*
- ❑ *PCI Local Bus Specification, Revision 3.0*

(See Section 1.3 for additional specification revision details.)

Applicable Device/Port Types

This test is run on all device/port types.

Starting Configuration

This test is run with functions starting in the D0-Uninitialized state following the standard initialization procedures in Section 2.1.1. Testing in additional Dx states may be optionally supported. ASPM states of the upstream and downstream ports are disabled for this test.

Overview of Test Steps

Test software performs the following steps:

1. Configure the function under test following the procedure described in Section 2.1.1.
2. A byte is read from location 3Dh (Interrupt Pin register) in configuration space for the function under test.
3. If the Interrupt Pin register is non-zero, the function must support at least either MSI or MSI-X interrupts and must implement the necessary MSI or MSI-X capability structure (only the MSI-X Capability structure will be tested here).
4. Examine the Capability ID field for each of the function's Capabilities. Determine how many instances of the Capability ID of 11h (MSI-X Capability) are found. If more than one is found, the test terminates with a failure.
5. If a Capability ID of 11h is found for a capability, the following checks are performed on the MSI-X capability structure.
6. A WORD is read from offset 00h in the MSI capability structure (Capability ID field, Next Capability Pointer field)
 - a. The Next Capability Pointer field must be 00h or greater than 3Fh and the lower 2 bits of this field must be 00b.
7. One DWORD is read from offset 04h (Table Offset/Table BIR) in the MSI-X capability structure.
8. The Table BIR field must return a valid value:
 - a. For a function with a Type 0 Configuration Space header valid values are (0, 1, 2, 3, 4, or 5).
 - b. For a function with a Type 1 Configuration Space header valid values are (0, 1).
 - c. The BAR pointed to by Table BIR is read and bit 0 of that BAR must be 0 (Memory BAR).
9. One DWORD is read from offset 08h (PBA Offset/PBA BIR) in the MSI-X capability structure.

10. The PBA BIR field must return a valid value:
 - a. For a function with a Type 0 Configuration Space header valid values are (0, 1, 2, 3, 4, or 5).
 - b. For a function with a Type 1 Configuration Space header valid values are (0, 1).
 - c. The BAR pointed to by PBA BIR is read and bit 0 of that BAR must be 0 (Memory BAR).
11. The following register field characteristic checks are performed:

MSI-X Capability Header (Offset 00h) — WORD

- | | |
|----------------------------|----|
| a. Capability ID | RO |
| b. Next Capability Pointer | RO |

Message Control Register (Offset 02h) — WORD

- | | |
|-------------------|---------|
| a. Table Size | RO |
| b. Reserved_13-11 | RO-Zero |
| c. Function Mask | RW |
| d. MSI-X Enable | RW |

Table Offset/Table BIR Register (Offset 04h) — DWORD

- | | |
|-----------------|----|
| a. Table BIR | RO |
| b. Table Offset | RO |

PBA Offset/PBA BIR (Offset 08h) — DWORD

- | | |
|---------------|----|
| a. PBA BIR | RO |
| b. PBA Offset | RO |

Message Address ([table] + ([n-1] * 16) + 00h) — DWORD

- | | |
|-------------------------|------------------|
| a. Message Address_1-0 | RW or
RO-Zero |
| b. Message Address_31-2 | RW |

Note: Testing this requires programming the designated Memory BAR and enabling Memory space. The value of [table] is the BAR contents (pointed to by Table BIR field) plus the offset (value in the Table Offset field multiplied by 8). All values of [n-1] between 0 and the value given in the Table Size field must be tested.

Message Upper Address ([table] + ([n-1] * 16) + 04h) — DWORD

- | | |
|--------------------------|----|
| a. Message Upper Address | RW |
|--------------------------|----|

Note: Testing this requires programming the designated Memory BAR and enabling Memory space. The value of [table] is the BAR contents (pointed to by Table BIR field) plus the offset (value in the Table Offset field multiplied by 8). All values of [n-1] between 0 and the value given in the Table Size field must be tested.

Message Data ([table] + ([n-1] * 16) + 08h) — DWORD

- | | |
|-----------------|----|
| a. Message Data | RW |
|-----------------|----|

Note: Testing this requires programming the designated Memory BAR and enabling Memory space. The value of [table] is the BAR contents (pointed to by Table BIR field) plus the offset (value in the Table Offset field multiplied by 8). All values of [n-1] between 0 and the value given in the Table Size field must be tested.

Vector Control ([table] + ([n-1] * 16) + 0Ch) — DWORD

- | | |
|---------------|---------|
| a. Mask Bit | RW |
| b. RsvdP_15-1 | RO-Zero |

- | | |
|--|---------------|
| c. ST Lower
(if function implements a TPH Requester Capability
and ST Table Location is 10b
and [n-1] is less than or equal to ST Table Size)
(otherwise) | RW
RO-Zero |
| d. ST Upper
(if function implements a TPH Requester Capability
and ST Table Location is 10b
and Extended TPH Requester Support is 1
and [n-1] is less than or equal to ST Table Size)
(otherwise) | RW
RO-Zero |

Note: Testing this requires programming the designated Memory BAR and enabling Memory space. The value of [table] is the BAR contents (pointed to by Table BIR field) plus the offset (value in the Table Offset field multiplied by 8). All values of [n-1] between 0 and the value given in the Table Size field must be tested.

Pending Bit Array ([ptable] + ([n-1] * 16) + 0Ch) — 2 DWORDS

- | | |
|-------------------------------|------------------|
| a. Pending Bits
(all bits) | RW or
RO-Zero |
|-------------------------------|------------------|

Note: Testing this requires programming the designated Memory BAR and enabling Memory decoding. The value of [ptable] is the BAR contents (pointed to by PBA BIR field) plus the offset (value in the PBA Offset field multiplied by 8). All values of [n-1] between 0 and the value given in the Table Size field must be tested.

12. The following default value checks are performed:

Message Control Register Default Value (Offset 02h) — WORD

- | | |
|------------------|---|
| a. Function Mask | 0 |
| b. MSI-X Enable | 0 |

Message Address Default Value ([table] + ([n-1] * 16) + 00h) — DWORD

- | | |
|------------------------|-----|
| a. Message Address_1-0 | 00b |
|------------------------|-----|

Note: Testing this requires programming the designated Memory BAR and enabling Memory space. The value of [table] is the BAR contents (pointed to by Table BIR field) plus the offset (value in the Table Offset field multiplied by 8). All values of [n-1] between 0 and the value given in the Table Size field must be tested.

Vector Control Default Value ([table] + ([n-1] * 16) + 0Ch) — DWORD

- | | |
|-------------|-----|
| a. Mask Bit | 1 |
| b. ST Lower | 00h |
| c. ST Upper | 00h |

Note: Testing this requires programming the designated Memory BAR and enabling Memory decoding. The value of [table] is the BAR contents (pointed to by Table BIR field) plus the offset (value in the Table Offset field multiplied by 8). All values of [n-1] between 0 and the value given in the Table Size field must be tested.

Pending Bit Array Default Value ([ptable] + ([n-1] * 8) + 00h) — 2 DWORDS

- a. Pending Bits
(all bits)

0

Note: Testing this requires programming the designated Memory BAR and enabling Memory decoding. The value of [ptable] is the BAR contents (pointed to by PBA BIR field) plus the offset (value in the PBA Offset field multiplied by 8). All values of [n-1] between 0 and the value given in the Table Size field must be tested.

13. For functions under test that have a link, the test is run at each of the following link speeds:
- 2.5 GT/s (for devices that are capable). See Section 2.1.2.12 for details.
 - 5.0 GT/s (for devices that are capable). See Section 2.1.2.13 for details.
 - 8.0 GT/s (for devices that are capable). See Section 2.1.2.14 for details.

The test *fails* if:

- ☐ More than one MSI-X capability structure is present.
- ☐ A non-zero Next Capability Pointer field value is less than or equal to 3Fh, or the lower 2 bits are non-zero.
- ☐ The Table BIR field has a reserved value.
- ☐ The Table BIR field points to an I/O Space BAR.
- ☐ The PBA BIR field has a reserved value.
- ☐ The PBA BIR field value points to an I/O Space BAR.
- ☐ Any of the register field characteristic tests fail.
- ☐ Any of the default value tests fail.

2.2.22. TD_1_18 Base Address Registers

The test verifies that the function under test implements all BAR registers as defined in the relevant specifications.

Relevant Specifications

- ☐ *PCI Express Base Specification*
- ☐ *PCI Local Bus Specification, Revision 3.0*
- ☐ *Single Root I/O Virtualization and Sharing Specification*

(See Section 1.3 for additional specification revision details.)

Applicable Device/Port Types

This test is run on all device/port types.

Starting Configuration

This test is run with functions starting in the D0-Uninitialized state following the standard initialization procedures in Section 2.1.1. Testing in additional Dx states may be optionally supported. ASPM states of the upstream and downstream ports are disabled for this test.

Overview of Test Steps

Test software performs the following steps:

1. Configure the function under test following the procedure described in Section 2.1.1.
2. Starting at location 10h the next 6 DWORDs (for a function with a Type 0 Configuration Space Header), or next 2 DWORDs (for a function with a Type 1 Configuration Space Header), are read. These are the BAR0, BAR1 (all functions), BAR2 to BAR5 (for a function with a Type 0 Configuration Space Header) registers. (Some BAR registers may occupy 2 DWORDs if they are 64 bit BARs.)
 - a. For a function with a Type 0 Configuration Space header, location 24h (BAR 5) must not have both bit 0 = 0 (Memory Space BAR) and Type field = 10b (64 bit addressing).
 - b. For a function with a Type 1 Configuration Space header, location 14h (BAR 1) must not have both bit 0 = 0 (Memory Space BAR) and Type field = 10b (64 bit addressing).
3. For a VF, for each BAR test software performs the following test:
 - a. Test software writes FFFF FFFFh to the BAR and then reads back the same register. The BAR register must return 0000 0000h.
 - b. For a VF, skip to step 9.
4. For each BAR register test software writes FFFF FFFFh to the BAR and then reads back the same register. If it returns 0000 0000h, then this is an empty BAR. Empty BARs are excluded from further testing.
5. For each non-empty BAR register test software checks bit 0 (Space Indicator). If the Space Indicator indicates an I/O Space BAR, then perform the tests in step 6. If the Space Indicator indicates a Memory Space BAR, then perform the tests in step 7.
6. For I/O Space BARs only:
 - a. Bit 1 must be zero.
 - b. Test software writes FFFF FFFFh to the BAR and then reads back the same register. Bit 8 must return 1 (I/O BAR must claim 256 Bytes or less).
 - c. Starting at bit 2 determine the least significant bit value [LSB] that is set to 1. [LSB] must be less than 9. All bits from 31 to [LSB] must also be 1. This same [LSB] value is used in subsequent tests of this same BAR.
 - d. Test software writes 5555 5555h to the BAR and then reads back the same register. Bits 31-[LSB] must return the written value. Bits [LSB]-2 must return all 0.
 - e. Test software writes AAAA AAAAh to the BAR and then reads back the same register. Bits 31-[LSB] must return the written value. Bits [LSB]-2 must return all 0.
 - f. Test software writes CCCC CCCCh to the BAR and then reads back the same register. Bits 31-[LSB] must return the written value. Bits [LSB]-2 must return all 0.
 - g. Test software writes 0000 0000h to the BAR and then reads back the same register. Bits 31-[LSB] must return the written value. Bits [LSB]-2 must return all 0.
 - h. If the function under test is a PCI/PCI-X to PCI Express Bridge or a PCI Express to PCI/PCI-X Bridge, then a test failure is recorded, but testing continues.

7. For MEMORY Space BARs only:
 - a. For all device types (except Legacy Endpoints and Root Complex Integrated Endpoints) if the Prefetchable bit is 1, then the Type field must be 10b (indicating only 64 bit addressing is supported).
 - b. For all device types (except Legacy Endpoints and Root Complex Integrated Endpoints) if the Prefetchable bit is 0, then the Type field must be 10b or 00b (indicating either 64 or 32 bit addressing is supported).
 - c. For Legacy Endpoints and Root Complex Integrated Endpoints the Type field must be 10b or 00b (indicating either 64 or 32 bit addressing is supported).
 - d. Test software writes FFFF FFFFh to the BAR and then reads back the same register. Bits 6-4 must be 0 (MEMORY BAR size must be 128 Bytes or larger).
 - e. If Type field returns 00b, then starting at bit 4 determine the least significant bit value [LSB] that is set to 1. [LSB] must be less than or equal to 31. All bits from 31 to [LSB] must also be 1. This same [LSB] value is used in subsequent tests of this same BAR.
 - f. If Type field returns 10b, test software writes FFFF FFFFh to the immediate next higher Configuration Space location (this is the upper half of a 64 bit BAR) then starting at bit 4 of the lower 32 bit BAR determine the least significant bit value [LSB] that is set to 1. [LSB] must be less than or equal to 63. All bits from 63 to [LSB] must also be 1.
 - g. Test software writes 5555 5555h (for 32 bit BAR) or 5555 5555 5555 5555h (for 64 bit BAR) to the BAR and then reads back the same register. Bits 31-[LSB] (for 32 bit BAR) or Bits 63-[LSB] (for 64 bit BAR) must return the written value. Bits [LSB]-4 must return all 0.
 - h. Test software writes AAAA AAAAh (for 32 bit BAR) or AAAA AAAA AAAA AAAAh (for 64 bit BAR) to the BAR and then reads back the same register. Bits 31-[LSB] (for 32 bit BAR) or Bits 63-[LSB] (for 64 bit BAR) must return the written value. Bits [LSB]-4 must return all 0.
 - i. Test software writes CCCC CCCCh (for 32 bit BAR) or CCCC CCCC CCCC CCCCh (for 64 bit BAR) to the BAR and then reads back the same register. Bits 31-[LSB] (for 32 bit BAR) or Bits 63-[LSB] (for 64 bit BAR) must return the written value. Bits [LSB]-4 must return all 0.
 - j. Test software writes 0000 0000h (for 32 bit BAR) or 0000 0000 0000 0000h (for 64 bit BAR) to the BAR and then reads back the same register. Bits 31-[LSB] (for 32 bit BAR) or Bits 63-[LSB] (for 64 bit BAR) must return the written value. Bits [LSB]-4 must return all 0.
8. The following register field characteristic checks are performed for each non-empty BAR (where the number of BARs depends on the Configuration Space Header type and the BAR size):

I/O BAR (n) Register — DWORD

(for non-VFs)

- | | |
|----------|---------|
| a. bit 0 | RO |
| b. bit 1 | RO-Zero |

Memory BAR (n) (lower 32) Register — DWORD

(for non-VFs)

- | | |
|-----------------|----|
| a. bit 0 | RO |
| b. Type | RO |
| c. Prefetchable | RO |

9. For functions under test that have a link, the test is run at each of the following link speeds:
 - a. 2.5 GT/s (for devices that are capable). See Section 2.1.2.12 for details.
 - b. 5.0 GT/s (for devices that are capable). See Section 2.1.2.13 for details.
 - c. 8.0 GT/s (for devices that are capable). See Section 2.1.2.14 for details.

The test *fails* if:

- ☐ The highest location that can contain a BAR reports it is the lower 32 bits of a 64 bit BAR
- ☐ A VF has a non-zero BAR.
- ☐ An I/O Space BAR claims more than 256 Bytes of I/O space per I/O Space BAR.
- ☐ An I/O Space BAR does not return contiguous 1's across all implemented address bits.
- ☐ An I/O Space BAR does not implement RW bits across all implemented address bits.
- ☐ A Memory Space BAR does not contain a valid Type field value.
- ☐ A device type that is not a Legacy Endpoint or a Root Complex Integrated Endpoint has a Memory BAR that is Prefetchable and does not support 64 bit addressing.
- ☐ A Memory Space BAR claims less than 128 Bytes of memory space.
- ☐ A Memory Space BAR does not return contiguous 1's across all implemented address bits.
- ☐ A Memory Space BAR does not implement RW bits across all implemented address bits.
- ☐ Any of the register field characteristic tests fail.

2.2.23. TD_1_19 Multi-Function Virtual Channel Extended Capability Structure

The test verifies that if a function under test reports a PCI Express Multi-Function Virtual Channel Capability structure, it is implemented as defined in the relevant specifications.

Relevant Specifications

- ☐ *PCI Express Base Specification*
- ☐ *Single Root I/O Virtualization and Sharing Specification*

(See Section 1.3 for additional specification revision details.)

Applicable Device/Port Types

This test is run on all device/port types.

This test is run on any RCRB associated with the function under test.

Starting Configuration

This test is run with functions starting in the D0-Uninitialized state following the standard initialization procedures in Section 2.1.1. Testing in additional Dx states may be optionally supported. ASPM states of the upstream and downstream ports are disabled for this test.

Overview of Test Steps:

Test software performs the following steps:

1. Configure the function under test following the procedure described in Section 2.1.1.

2. Examine the Extended Capability ID field for each of the detected Extended Capabilities. Determine how many instances of the Extended Capability ID of 0008h (Multi-Function Virtual Channel Extended Capability) are found. If more than one is found, the test terminates with a failure.
3. If the Extended Capability ID of 0008h is found in a VF, the test terminates with a failure.
4. If the Extended Capability ID of 0008h is found in an RCRB, the test terminates with a failure.
5. If an Extended Capability ID of 0008h is found for an extended capability the following checks are performed on that extended capability structure:
 6. The function reporting the MFVC Capability must be a Device type associated with an upstream port.
 7. The function reporting the MFVC Capability must be function 0, and the byte at location 0Eh (Header Type register) in Configuration Space must have bit 7 set to 1 (it must be part of a multi-function device).
 8. The Capability Version field must be 1h.
 9. The Next Capability Offset field must be 000h or greater than 0FFh and the lower 2 bits of this field must be 00b.
10. Test software reads the DWORD at offset 04h (Port VC Capability Register 1) and performs the following checks:
 - a. The Low Priority Extended VC Count field is less than or equal to the Extended VC Count field value.
 - b. The Reference Clock field must be 00b.
11. Test software reads the DWORD from offset 08h (Port VC Capability Register 2 register) and performs the following checks:
 - a. No reserved bits (4-7) in the VC Arbitration Capability field may be set.
 - b. The VC Arbitration Capability field must not be zero if the Low Priority Extended VC Count field value is non-zero.
 - c. The VC Arbitration Capability field must be zero if the Low Priority Extended VC Count field value is zero.
 - d. The VC Arbitration Table Offset field must not be zero if any of bits 1, 2, or 3 of the VC Arbitration Capability field are set.
12. Test software reads the DWORD from offset 0Ch (Port VC Control register and Port VC Status register) and performs the following checks:
 - a. The VC Arbitration Table Status field reads zero.
13. If the default VC Arbitration Select field value is a WRR based arbitration method test software reads the default values in the VC Arbitration table. These values must all be zero.
14. Test software writes each of the supported data values indicated by the VC Arbitration Capability field to the VC Arbitration Select field and verifies the value is read back.
15. For each supported WRR arbitration field mechanism, test software tests loading the table by writing a table of all zeros of the appropriate size to the indicated VC Arbitration table offset. Test software then checks that the VC Arbitration Table Status field returns 1. Test software then writes 1 to the Load VC Arbitration Table field. Test software then reads the VC

Arbitration Table Status field repeatedly until it returns 0. If it does not return 0 within 1 second, the test fails.

16. Test software repeats step 15 using all VC entries for the highest supported VC instead of zero.
17. Test software verifies that VC Resource Control, VC Resource Status, and VC Resource Capability registers are present for each VC supported by the function (as indicated by the Extended VC Count field value).
18. For each set of VC Resource Control, VC Resource Status, and VC Resource Capability registers as reported in the Extended VC Count field value, test software performs the following checks:
 - a. The Function Arbitration Capability field must not have bit 6 or 7 set (reserved).
 - b. The TC/VC Map field must be FFh for the first VC resource (VC0) by default. For all other VCs it must be 00h by default. Test software writes each valid TC combination to the field and ensures that the same value can be read back, with the exception of bit 0 which is RO=1 for the first VC resource (VC0) and RO=0 for all other VCs.
 - c. The Load Function Arbitration Table field must be zero.
 - d. The Function Arbitration Select field: Test software writes each supported data value (based on the Function Arbitration Capability field value) and ensures that the same value is read back.
 - e. The VC-ID field: Test software ensures that the value is hard-wired to 000b for the first VC resource (VC0).
 - f. The VC Enable field: Test software checks that this bit is hardwired to one for the first VC resource (VC0).
 - g. The Function Arbitration Table Status field must be zero.
19. For each supported WRR based Function Arbitration scheme test software writes a table with all values of the minimum Function number. After the first write to the table test software checks that the Function Arbitration Table Status field returns 1. Test software writes 1 to the Load Function Arbitration field. Test software then reads the Function Arbitration Table Status field repeatedly until it returns 0. If it does not return 0 within 1 second, the test fails.
20. Test software repeats step 19 with a valid arbitration table containing all functions in the multi-function device.
21. The following register field characteristic checks are performed:

MFVC Extended Capability Header (Offset 00h) — DWORD

- | | |
|---------------------------------------|----|
| a. PCI Express Extended Capability ID | RO |
| b. Capability Version | RO |
| c. Next Capability Offset | RO |

Port VC Capability Register 1 (Offset 04h) — DWORD

- | | |
|--|---------|
| a. Extended VC Count | RO |
| b. RsvdP_3 | RO-Zero |
| c. Low Priority Extended VC Count | RO |
| d. RsvdP_7 | RO-Zero |
| e. Reference Clock | RO |
| f. Function Arbitration Table Entry Size | RO |
| g. RsvdP_31-12 | RO-Zero |

Port VC Capability Register 2 (Offset 08h) — DWORD

- | | |
|------------------------------|----|
| a. VC Arbitration Capability | RO |
|------------------------------|----|

- b. RsvdP_23-8 RO-Zero
- c. VC Arbitration Table Offset RO

Port VC Control Register (Offset 0Ch) — WORD

- a. Load VC Arbitration Table RO-Zero
- b. VC Arbitration Select
 - (if more than one VC in the LPVC group is not enabled) RW
 - (if more than one VC in the LPVC group is enabled) RO
 - (only those values in VC Arbitration Capability are written)
- c. RsvdP_15-4 RO-Zero

Port VC Status Register (Offset 0Eh) — WORD

- a. VC Arbitration Table Status RO
- b. RsvdZ_15-1 RO-Zero

22. For each VC Resource register set value [n] (given by value in Extended VC Count), the following register field characteristic checks and register field default value checks are performed:

VC Resource Capability Register (n) (Offset 10h + n * 0Ch) — DWORD

- a. Function Arbitration Capability RO
- b. RsvdP_15-8 RO-Zero
- c. Maximum Time Slots
 - (for non-FLR testing) HwInit
 - (for FLR testing) RO
- d. RsvdP_23 RO-Zero
- e. Function Arbitration Table Offset RO

VC Resource Control Register (n) (Offset 14h + n * 0Ch) — DWORD

- a. TC/VC Map
 - (for VC0: bit 0) RO-Ones
 - (for non-VC0: bit 0) RO-Zero
- b. (bits 7-1) RW
- c. RsvdP_15-8 RO-Zero
- d. Load Function Arbitration Table RO-Zero
- e. Function Arbitration Select
 - (only those values in Port Arbitration Capability are written) RW
- f. RsvdP_23-20 RO-Zero
- g. VC ID
 - (for VC0) RO-Zero
 - (for all non VC0) RW
- h. RsvdP_30-27 RO-Zero
- i. VC Enable
 - (for VC0) RO
 - (for all non-VC0) RW

Reserved Register (n) (Offset 18h + n * 0Ch) — WORD

- a. RsvdP_15-0 RO-Zero

VC Resource Status Register (n) (Offset 1Ah + n * 0Ch) — WORD

- | | |
|--------------------------------------|---------|
| a. Function Arbitration Table Status | RO |
| b. VC Negotiation Pending | RO |
| c. RsvdZ_15-2 | RO-Zero |

VC Resource Control Register (n) Default Value (Offset 14h + n * 0Ch) — DWORD

- | | |
|-------------------|-----|
| a. TC/VC Map | |
| (for VC0) | FFh |
| (for all non VC0) | 00h |
| b. VC Enable | |
| (for VC0) | 1 |
| (for all non-VC0) | 0 |

23. For functions under test that have a link, the test is run at each of the following link speeds:

- 2.5 GT/s (for devices that are capable). See Section 2.1.2.12 for details.
- 5.0 GT/s (for devices that are capable). See Section 2.1.2.13 for details.
- 8.0 GT/s (for devices that are capable). See Section 2.1.2.14 for details.

24. If the device type is Root Port or Root Complex Integrated Endpoint, then the test is repeated using each RCRB associated with the function under test, or associated with an RCRB that is itself associated with the function under test. See Section 2.1.2.17 for details. All offsets in the tests are now relative to the RCRB's base address and all cycles are now memory space accesses.

The test *fails* if:

- ☐ More than one MFVC capability structure is present.
- ☐ A VF contains a MFVC capability structure.
- ☐ An RCRB contains a MFVC capability structure
- ☐ A MFVC capability structure is present anywhere except in Function 0 in the Upstream Port of a multi-function device.
- ☐ The Capability Version field does not report 1h.
- ☐ The Next Capability Offset field does not report 000h or a value greater than 0FFh or the lower two bits are not 00b.
- ☐ The Low Priority Extended VC Count field is greater than the Extended VC Count field (Port VC Capability Register 1).
- ☐ The Reference Clock field is not zero (Port VC Capability Register 1).
- ☐ The VC Arbitration Capability field contains a reserved value and the Low Priority Extended VC Count field is non-zero.
- ☐ The VC Arbitration Capability field contains a zero value and the Low Priority Extended VC Count field is non-zero.
- ☐ The VC Arbitration Table Offset field is zero when WRR based VC Arbitration Capabilities are supported.
- ☐ The VC Arbitration Table Status field reads non-zero by default.
- ☐ The default VC Arbitration Select value is a WRR based arbitration method and the default VC Arbitration table values are not all zero.

- ❑ VC Resource Control, VC Resource Status, and VC Resource Capability registers are not present for each VC supported by the function (as indicated by Extended VC Count field).
- ❑ A valid Function Arbitration Select field value or Function Arbitration table cannot be written.
- ❑ Any of the described VC Resource Control and VC Resource Status register value checks are not met.
- ❑ Any of the register field characteristic tests fail.
- ❑ Any of the default value tests fail.

2.2.24. TD_1_52 Vendor Specific Capability Structure

The test verifies that if a function under test reports a Vendor Specific Capability structure, it is implemented as defined in the relevant specifications.

Relevant Specifications

- ❑ *PCI Local Bus Specification*

(See Section 1.3 for additional specification revision details.)

Applicable Device/Port Types

This test is run on all device/port types.

Starting Configuration

This test is run with functions starting in the D0-Uninitialized state following the standard initialization procedures in Section 2.1.1. Testing in additional Dx states may be optionally supported. ASPM states of the upstream and downstream ports are disabled for this test.

Overview of Test Steps:

Test software performs the following steps:

1. Configure the function under test following the procedure described in Section 2.1.1.
2. Examine the Capability ID field for each of the function's Capabilities. Determine how many instances of the Capability ID of 09h (Vendor Specific Capability) are found.
3. If a Capability ID field of 09h is found for a capability the following checks are performed for each instance of that capability:
4. The Next Capability Pointer field must be 00h or greater than 3Fh and the lower 2 bits of this field must be 00b.
5. The Next Capability Pointer field must be a value less than the offset of this capability by at least 4 bytes or must be larger than the offset of this capability plus the VSC Length field value rounded to the next highest DWORD boundary.
6. The value for VSC Length field must be at least 03h (3 bytes).
7. If the VSC Length field is greater than 03h, test software reads each byte starting at offset 03h, and continuing until the byte length given by the value in the VSC Length field is reached. (This checks that Vendor Specific registers can be read safely. No check on the returned data value is done.)
8. The following register field characteristic checks are performed:

Vendor Specific Capability Header (Offset 00h) — WORD

- | | |
|----------------------------|----|
| a. Capability ID | RO |
| b. Next Capability Pointer | RO |

VSC Length (Offset 02h) — BYTE

RO

9. For functions under test that have a link, the test is run at each of the following link speeds:
 - a. 2.5 GT/s (for devices that are capable). See Section 2.1.2.12 for details.
 - b. 5.0 GT/s (for devices that are capable). See Section 2.1.2.13 for details.
 - c. 8.0 GT/s (for devices that are capable). See Section 2.1.2.14 for details.

The test *fails* if:

- ☐ A non-zero Next Capability Pointer field is less than or equal to 3Fh, or the lower 2 bits are non-zero.
- ☐ The Next Capability Pointer field is not at least 4 less than the current capability offset or it is less than the current capability offset plus the VSC Length field value rounded to the next highest DWORD boundary.
- ☐ The VSC Length field is less than 3.
- ☐ Any of the register field characteristic tests fail.

2.2.25. TD_1_20 Vendor-Specific Extended Capability Structure

The test verifies that if a function under test reports a PCI Express Vendor-Specific Extended Capability structure, it is implemented as defined in the relevant specifications.

Relevant Specifications

- ☐ *PCI Express Base Specification*

(See Section 1.3 for additional specification revision details.)

Applicable Device/Port Types

This test is run on all device/port types.

This test is run on any RCRB associated with the function under test.

Starting Configuration

This test is run with functions starting in the D0-Uninitialized state following the standard initialization procedures in Section 2.1.1. Testing in additional Dx states may be optionally supported. ASPM states of the upstream and downstream ports are disabled for this test.

Overview of Test Steps:

Test software performs the following steps:

1. Configure the function under test following the procedure described in Section 2.1.1.
2. Examine the Extended Capability ID field for each of the detected Extended Capabilities. Determine how many instances of the Extended Capability ID of 000Bh (Vendor-Specific Extended Capability) are found.

3. If the Extended Capability ID of 000Bh is found in an RCRB the following checks are performed:
 - a. Examine the Extended Capability ID field for each of the RCRB's Extended Capabilities. Determine how many instances of the Extended Capability ID of 000Ah (RCRB Header Extended Capability) are found. If none are found, the test terminates with a failure.
4. If an Extended Capability ID of 000Bh is found for an extended capability the following checks are performed for each instance of that extended capability structure:
5. The Capability Version field must be 1h.
6. The Next Capability Offset field must be 000h or greater than 0FFh and the lower 2 bits of this field must be 00b.
7. The Next Capability Offset field must return a value less than the offset of this capability by at least 8 bytes or must be larger than the offset of this capability plus the VSEC Length field value rounded to the next highest DWORD boundary.
8. The value for the VSEC Length field must be at least 008h (8 bytes).
9. If the VSEC Length field is greater than 008h, test software reads each byte starting at offset 08h, and continuing until the byte length given by the value in the VSEC Length field is reached. (This checks that Vendor-Specific registers can be read safely. No check on the returned data value is done.)
10. The following register field characteristic checks are performed:

Vendor-Specific Capability Extended Capability Header (Offset 00h) — DWORD

- | | |
|---------------------------------------|----|
| a. PCI Express Extended Capability ID | RO |
| b. Capability Version | RO |
| c. Next Capability Offset | RO |

Vendor-Specific Header (Offset 04h) — DWORD

- | | |
|----------------|----|
| a. VSEC ID | RO |
| b. VSEC Rev | RO |
| c. VSEC Length | RO |

11. For functions under test that have a link, the test is run at each of the following link speeds:
 - a. 2.5 GT/s (for devices that are capable). See Section 2.1.2.12 for details.
 - b. 5.0 GT/s (for devices that are capable). See Section 2.1.2.13 for details.
 - c. 8.0 GT/s (for devices that are capable). See Section 2.1.2.14 for details.
12. If the device type is Root Port or Root Complex Integrated Endpoint, then the test is repeated using each RCRB associated with the function under test, or associated with an RCRB that is itself associated with the function under test. See Section 2.1.2.17 for details. All offsets in the tests are now relative to the RCRB's base address and all cycles are now memory space accesses.

The test *fails* if:

- ☐ An RCRB contains a Vendor Specific capability structure, but does not contain an RCRB Header capability structure.
- ☐ The Capability Version field does not report 1h.
- ☐ The Next Capability Offset field does not report 000h or a value greater than 0FFh or the lower two bits are not 00b.
- ☐ The Next Capability Offset field is not at least 8 less than the current capability offset or it is less than the current capability offset plus the VSEC Length field value rounded to the next highest DWORD boundary.
- ☐ The VSEC Length field is less than 8.
- ☐ Any of the register field characteristic tests fail.

2.2.26. TD_1_21 BIST Register

This test verifies that the function under test implements the BIST register as defined in the relevant specifications. (This tests both the case where HW BIST is implemented and the case where HW BIST is not implemented.)

Relevant Specifications

- ☐ *PCI Local Bus Specification*
- ☐ *Single Root I/O Virtualization and Sharing Specification*

(See Section 1.3 for additional specification revision details.)

Applicable Device/Port Types

This test is run on all device/port types.

Starting Configuration

This test is run with functions starting in the D0-Uninitialized state following the standard initialization procedures in Section 2.1.1. Testing in additional Dx states may be optionally supported. ASPM states of the upstream and downstream ports are disabled for this test.

Overview of Test Steps

Test software performs the following steps:

1. Configure the function under test following the procedure described in Section 2.1.1.
2. Read the byte from location 0Fh (BIST register) in configuration space.
3. If the BIST Capable field is 0, the following checks are performed:
 - a. Write a 1 to Bit 6 (Start BIST field) to see it has no effect.
 - b. Immediately read the BIST register and check the following:
 - i. Bit 6 (Start BIST field) must be zero.
 - ii. Bits 3-0 (Completion Code field) must return zero.
 - iii. Bits 5-4 (Reserved field) must return zero.

4. For non-VFs, if the BIST Capable field is 1, the following checks are performed:
 - a. Write a 1 to Bit 6 (Start BIST field) to invoke BIST.
 - b. After 2 seconds, read the BIST register and check the following:
 - i. Bit 6 (Start BIST field) must be zero.
 - ii. Bits 3-0 (Completion Code field) must return zero.
 - iii. Bits 5-4 (Reserved field) must return zero.

5. The following register field characteristic checks are performed:

BIST Register (Location 0Fh) — BYTE

- | | |
|------------------------|---------|
| a. Completion Code | |
| (if BIST Capable is 1) | RO |
| (if BIST Capable is 0) | RO-Zero |
| b. Reserved_5-4 | RO-Zero |
| c. Start BIST | |
| (if BIST Capable is 0) | RO-Zero |
| d. BIST Capable | |
| (for VFs) | RO-Zero |
| (otherwise) | RO |

6. The following default value checks are performed:

BIST Register Default Value (Location 0Fh) — BYTE

- | | |
|--------------------|-------|
| a. Completion Code | 0000b |
| b. Start BIST | 0 |

7. For functions under test that have a link, the test is run at each of the following link speeds:
 - a. 2.5 GT/s (for devices that are capable). See Section 2.1.2.12 for details.
 - b. 5.0 GT/s (for devices that are capable). See Section 2.1.2.13 for details.
 - c. 8.0 GT/s (for devices that are capable). See Section 2.1.2.14 for details.

The test *fails* if:

- ☐ If the BIST Capable field is 0 and immediately after writing 1 to the Start BIST field, reading the BIST register returns non-zero values for the following fields: Completion Code, Reserved, or Start BIST.
- ☐ If the BIST Capable field is 1 and 2 seconds after writing 1 to the Start BIST field, reading the BIST register returns non-zero values for the following fields: Completion Code, Reserved, or Start BIST.
- ☐ Any of the register field characteristic tests fail.
- ☐ Any of the default value tests fail.

2.2.27. TD_1_22 Slot Numbering Capability Structure

The test verifies that if the function under test reports a Slot Numbering capability, it is implemented as defined in the relevant specifications. (Since this capability may only be present in a PCI Express to PCI/PCI-X Bridge, the test also checks that this capability is not present in other device types.)

Relevant Specifications

- ❑ *PCI Express to PCI/PCI-X Bridge Specification, Revision 1.0*
- ❑ *PCI-to-PCI Bridge Architecture Specification, Revision 1.2*

(See Section 1.3 for additional specification revision details.)

Applicable Device/Port Types

This test is run on all device/port types.

Starting Configuration

This test is run with functions starting in the D0-Uninitialized state following the standard initialization procedures in Section 2.1.1. Testing in additional Dx states may be optionally supported. ASPM states of the upstream and downstream ports are disabled for this test.

Overview of Test Steps

Test software performs the following steps:

1. Configure the function under test following the procedure described in Section 2.1.1.
2. Examine the Capability ID field for each of the function's Capabilities. Determine how many instances of the Capability ID of 04h (Slot Numbering Capability) are found. If more than one is found, the test terminates with a failure.
3. If a Capability ID of 04h is found for a capability, the following checks are performed.
4. The Slot Numbering capability is only valid for a device type of PCI Express to PCI/PCI-X Bridge (a device with a PCI slot). For other device types, this capability must not be present or the test terminates with failure.
5. The Next Capability Pointer field must be 00h or greater than 3Fh and the lower 2 bits of this field must be 00b.
6. Read the WORD from offset 02h (Expansion Slot and Chassis Number registers). If Bit 5 (First in Chassis field) is set, the Chassis Number register must be non zero and unique. (For the purposes of this test, a non-zero value is assumed unique.)

7. The following register field characteristic checks are performed:

Slot Numbering Capability Header (Offset 00h) — WORD

- | | |
|----------------------------|----|
| a. Capability ID | RO |
| b. Next Capability Pointer | RO |

Expansion Slot Register (Offset 02h) — BYTE

- | | |
|--|---------|
| a. Expansion Slots Provided
(for non-FLR testing) | HwInit |
| (for FLR testing) | RO |
| b. First in Chassis
(for non-FLR testing) | HwInit |
| (for FLR testing) | RO |
| c. Reserved_7-6 | RO-Zero |

Chassis Number Register (Offset 03h) — BYTE

- | | |
|-------------------|----|
| a. Chassis Number | RW |
|-------------------|----|

8. For functions under test that have a link, the test is run at each of the following link speeds:
- 2.5 GT/s (for devices that are capable). See Section 2.1.2.12 for details.
 - 5.0 GT/s (for devices that are capable). See Section 2.1.2.13 for details.
 - 8.0 GT/s (for devices that are capable). See Section 2.1.2.14 for details.

The test *fails* if:

- ☐ More than one Slot Numbering capability structure is present.
- ☐ A non-zero Next Capability Pointer field is less than or equal to 3Fh, or the lower 2 bits are non-zero.
- ☐ For devices other than a PCI Express to PCI/PCI-X Bridge, the Slot Numbering capability is present.
- ☐ For PCI Express to PCI/PCI-X Bridge devices:
 - The Expansion Slots Provided field is zero
 - The Chassis Number register is zero and cannot be written with a non-zero value.
- ☐ Any of the register field characteristic tests fail.

2.2.28. TD_1_53 SSID/SSVID Capability Structure

The test verifies that if the function under test reports a SSID/SSVID capability, it is implemented as defined in the relevant specifications. (Since this capability may only be present in a device with a Type 1 Configuration Space Header, the test also checks that this capability is not present in other Configuration Space Header types.)

Relevant Specifications

- ☐ *PCI-to-PCI Bridge Architecture Specification, Revision 1.2*

(See Section 1.3 for additional specification revision details.)

Applicable Device/Port Types

This test is run on all device/port types.

Starting Configuration

This test is run with functions starting in the D0-Uninitialized state following the standard initialization procedures in Section 2.1.1. Testing in additional Dx states may be optionally supported. ASPM states of the upstream and downstream ports are disabled for this test.

Overview of Test Steps

Test software performs the following steps:

1. Configure the function under test following the procedure described in Section 2.1.1.
2. Examine the Capability ID field for each of the function's Capabilities. Determine how many instances of the Capability ID of 0Dh (SSID/SSVID Capability) are found. If more than one is found, the test terminates with a failure.
3. If a Capability ID of 0Dh is found for a capability, the following checks are performed.
4. Read a byte from location 0Eh (Header Type register). If the value in bits 6-0 is not 000 0001b the test terminates with failure (SSID/SSVID Capability may only be present in a device with a Type 1 Configuration Space Header).
5. The Next Capability Pointer field must be 00h or greater than 3Fh and the lower 2 bits of this field must be 00b.
6. Read 3 bytes from location 09h (Class Code register). (Base Class is byte from location 0Bh, and Sub-Class is byte from location 0Ah.)
7. Read the WORD from offset 04h (SSVID register) and perform the following checks:
 - a. The Subsystem Vendor ID field must return a valid value assigned by the PCI-SIG. (For the purposes of this test program, a valid value is any value other than 0000h, or FFFFh, or 0001h.)
8. The following register field characteristic checks are performed:

SSID/SSVID Capability Header (Offset 00h) — WORD

- | | |
|----------------------------|----|
| a. Capability ID | RO |
| b. Next Capability Pointer | RO |

Reserved Register (Offset 02h) — WORD

- | | |
|------------------|---------|
| a. Reserved_15-0 | RO-Zero |
|------------------|---------|

SSVID Register (Offset 04h) — WORD

- | | |
|----------|----|
| a. SSVID | RO |
|----------|----|

SSID Register (Offset 06h) — WORD

- | | |
|---------|----|
| a. SSID | RO |
|---------|----|

9. For functions under test that have a link, the test is run at each of the following link speeds:
 - a. 2.5 GT/s (for devices that are capable). See Section 2.1.2.12 for details.
 - b. 5.0 GT/s (for devices that are capable). See Section 2.1.2.13 for details.
 - c. 8.0 GT/s (for devices that are capable). See Section 2.1.2.14 for details.

The test *fails* if:

- ☐ More than one SSID/SSVID capability structure is present.
- ☐ A non-zero Next Capability Pointer field is less than or equal to 3Fh, or the lower 2 bits are non-zero.
- ☐ For functions with a Configuration Space Header type that is not Type 1, the SSID/SSVID capability is present.
- ☐ The SSVID field returns 0000h, or FFFFh, or 0001h.
- ☐ Any of the register field characteristic tests fail.

2.2.29. TD_1_23 PCI Next Capability Pointer Register

The test verifies that the function under test implements the Next Capability Pointer registers fields as defined in the relevant specifications.

Relevant Specifications

- ☐ *PCI Express Base Specification*
- ☐ *PCI Local Bus Specification, Revision 3.0*

(See Section 1.3 for additional specification revision details.)

Applicable Device/Port Types

This test is run on all device/port types.

Starting Configuration

This test is run with functions starting in the D0-Uninitialized state following the standard initialization procedures in Section 2.1.1. Testing in additional Dx states may be optionally supported. ASPM states of the upstream and downstream ports are disabled for this test.

Overview of Test Steps

Test software performs the following steps:

1. Configure the function under test following the procedure described in Section 2.1.1.
2. Read the WORD from location 06h (Status register) and if the Capabilities List is set, the following checks are performed.
3. Read the byte from location 34h (Capabilities Pointer register) in the Configuration Space and perform the following checks:
 - a. The lower two bits must be zero (00b).
 - b. The value read, if non-zero, must be greater than 3Fh and less than FDh.
 - c. The value (with the lower two bits masked to 00b) is recorded in a table to check that subsequent Next Cap Pointers do not use the same value (creating a loop of pointers).
4. The current pointer value (with the lower two bits masked to 00b), if non-zero and if greater than 3Fh and less than FFh, is used to retrieve the Next Cap Pointer field of the capability that the capability pointer is indicating. Otherwise if the pointer is the end of list, skip to step 7.
5. If the Next Cap Pointer value is not zero the following checks are performed:
 - a. The lower two bits must be zero (00b).
 - b. The value read must be greater than 3Fh and less than FDh.

- c. The value (with the lower two bits masked to 00b) is recorded in a table to check that subsequent Next Cap Pointers do not use the same value (creating a loop of pointers).
- 6. Test software repeats steps 4-5 until:
 - a. The Next Cap Pointer is zero, or
 - b. The Value is not greater than 3Fh and less than FDh, or
 - c. The current pointer value (with the lower two bits masked to 00b) was used by the Capabilities Pointer or a previous Next Cap Pointer (this stops any loop of pointers).
- 7. For functions under test that have a link, the test is run at each of the following link speeds:
 - a. 2.5 GT/s (for devices that are capable). See Section 2.1.2.12 for details.
 - b. 5.0 GT/s (for devices that are capable). See Section 2.1.2.13 for details.
 - c. 8.0 GT/s (for devices that are capable). See Section 2.1.2.14 for details.

The test *fails* if:

- ☐ The Capability Pointer registers cannot be read.
- ☐ Any capability pointer contains an invalid value (either the lower two bits are not 00b, or the value is not greater than 3Fh and less than FDh).
- ☐ The capability list contains a loop of pointers.

2.2.30. TD_1_24 PCI Express Next Extended Capability Pointer Register

The test verifies that the function under test implements the Next Extended Capability Pointer registers fields as defined in the relevant specifications.

Relevant Specifications

- ☐ *PCI Express Base Specification*

(See Section 1.3 for additional specification revision details.)

Applicable Device/Port Types

This test is run on all device/port types.

This test is run on any RCRB associated with the function under test.

Starting Configuration

This test is run with functions starting in the D0-Uninitialized state following the standard initialization procedures in Section 2.1.1. Testing in additional Dx states may be optionally supported. ASPM states of the upstream and downstream ports are disabled for this test.

Overview of Test Steps

Test software performs the following steps:

1. Configure the function under test following the procedure described in Section 2.1.1.
2. If testing Extended Configuration Space, read the DWORD from location 100h in configuration space.
3. If testing RCRB, read the DWORD from offset 000h in the RCRB (in memory space).

4. If testing Extended Configuration Space, then if the value is not 0000 0000h, the following checks are performed. If the value is 0000 0000h (no Extended Capabilities present) skip to step 9.
5. If testing RCRB, then if the value is not 000x FFFFh (where x indicates that bits 19-16 are don't cares), the following checks are performed. If the value is 000x FFFFh (no Extended Capabilities present) skip to step 9.
6. If the Next Capability Offset field value is not 000h, the following checks are performed. If the value is 000h (no further Extended Capabilities present), skip to step 9.
 - a. The lower two bits must be zero (00b).
 - b. The value read must be greater than 107h and less than FF9h.
 - c. The value (with the lower two bits masked to 00b) is recorded in a table to check that subsequent Next Capability Offsets do not use the same value (creating a loop of pointers).
7. The current pointer value (with the lower two bits masked to 00b), if non-zero and if greater than 107h and less than FF9h, is used to retrieve the Next Capability Offset of the extended capability that the extended capabilities pointer is indicating. Otherwise if the pointer is the end of list, skip to step 9.
8. Read the DWORD from the offset indicated in the current Next Capability Offset. Test software repeats steps 4-7 until:
 - a. The Next Capability Offset is 000h, or
 - b. The Value is not greater than 107h and less than FF9h, or
 - c. The current pointer value (with the lower two bits masked to 00b) was used by a previous Next Capability Offset (this stops any loop of pointers).
9. For functions under test that have a link, the test is run at each of the following link speeds:
 - a. 2.5 GT/s (for devices that are capable). See Section 2.1.2.12 for details.
 - b. 5.0 GT/s (for devices that are capable). See Section 2.1.2.13 for details.
 - c. 8.0 GT/s (for devices that are capable). See Section 2.1.2.14 for details.
10. If the device type is Root Port or Root Complex Integrated Endpoint, then the test is repeated using each RCRB associated with the function under test, or associated with an RCRB that is itself associated with the function under test. See Section 2.1.2.17 for details. All offsets in the tests are now relative to the RCRB's base address and all cycles are now memory space accesses.

The test *fails* if:

- ☐ The Extended Capability header registers cannot be read.
- ☐ Any Next Capability Offset contains an invalid value (either the lower two bits are not 00b, or the value is not greater than 107h and less than FF9h).
- ☐ The capability list contains a loop of pointers.

2.2.31. TD_1_25 Misc Type 0 Config Space Header Registers

The test verifies that if the function under test reports a Type 0 Configuration Space Header, it implements the Vendor ID, Device ID, Revision ID, Class Code, Header Type, Cardbus CIS Pointer, Subsystem ID, Subsystem Vendor ID, and Expansion ROM Base Address registers as defined in the relevant specifications.

Relevant Specifications

- ❑ *PCI Express Base Specification*
- ❑ *PCI Local Bus Specification, Revision 3.0*
- ❑ *Single Root I/O Virtualization and Sharing Specification*

(See Section 1.3 for additional specification revision details.)

Applicable Device/Port Types

This test is run on all device/port types implementing Type 0 Configuration Space Headers.

Starting Configuration

This test is run with functions starting in the D0-Uninitialized state following the standard initialization procedures in Section 2.1.1. Testing in additional Dx states may be optionally supported. ASPM states of the upstream and downstream ports are disabled for this test.

Overview of Test Steps

Test software performs the following steps:

1. Configure the function under test following the procedure described in Section 2.1.1.
2. Read a byte from location 0Eh (Header Type register). If the value in bits 6-0 is 000 0000b (Type 0 Configuration Space Header) the following checks are performed.
3. For a VF, Header Type bit 7 must return 0.
4. Read a WORD from location 00h (Vendor ID register) and perform the following checks:
 - a. For a VF, the Vendor ID field must return FFFFh.
 - b. For a non-VF, the Vendor ID field must return a valid value assigned by the PCI-SIG. (For the purposes of this test program, reads of the Vendor ID field must not return 0000h, FFFFh, or 0001h (CRS Software Visibility notification).)
5. Read 3 bytes from location 09h (Class Code register) and perform the following checks:
 - a. For device type of Root Complex Event Collector, the Base Class field must be 08h, the Sub-Class field must be 06h, and the Programming Interface field (byte at 09h) must be 00h.
6. Read a WORD from location 2Ch (Subsystem Vendor ID register) and perform the following checks:
 - a. If the function is neither Base Class 06h, Sub-Class 00h-04h nor Base Class 08h, Sub-Class 00h-03h, then the Subsystem Vendor ID field must return a valid value assigned by the PCI-SIG.
(For the purposes of this test program, a valid value is any value other than 0000h, FFFFh, or 0001h (CRS Software Visibility notification).)
7. Test software writes FFFF FFFFh to the Expansion ROM Base Address (location 30h for Type 0 Configuration Space Header) register and then reads back the same register and if it is zero the test skips to step 12, but if it is non-zero, the following checks are performed:
 - a. The function under test must not be a VF. If it is a VF, a failure is reported, but testing continues at step 12.
 - b. Bit 0 (Expansion ROM Enable field) must return 1.
 - c. Bit 24 must return 1 (ROM BAR must claim 16 MB or less).

- d. Starting at bit 11 determine the least significant bit value [LSB] that is set to 1. [LSB] must be less than 25. All bits from 31 to [LSB] must also be 1. This same [LSB] value is used in subsequent tests.
8. Test software writes 5555 5555h to the ROM BAR and then reads back the same register. Bits 31-[LSB] must return the written value. Bits [LSB]-11 must return all 0.
9. Test software writes AAAA AAAAh to the ROM BAR and then reads back the same register. Bits 31-[LSB] must return the written value. Bits [LSB]-11 must return all 0.
10. Test software writes CCCC CCCCh to the ROM BAR and then reads back the same register. Bits 31-[LSB] must return the written value. Bits [LSB]-11 must return all 0.
11. Test software writes 0000 0000h to the ROM BAR and then reads back the same register. Bits 31-[LSB] must return the written value. Bits [LSB]-11 must return all 0. Bit 0 (Expansion ROM Enable) must return 0.
12. The following register field characteristic checks are performed:

Vendor ID Register (Location 00h) — WORD

(for VFs)

RO-Ones

(otherwise)

RO

Device ID Register (Location 02h) — WORD

(for VFs)

RO-Ones

(otherwise)

RO

Revision ID Register (Location 08h) — BYTE

RO

Class Code Register (Location 09h) — 3 BYTES

a. Programming Interface

RO

b. Sub-Class

RO

c. Base Class

RO

Header Type Register (Location 0Eh) — BYTE

(for VFs)

RO-Zero

(otherwise)

RO

Cardbus CIS Pointer Register (Location 28h) — DWORD

(for VFs)

RO-Zero

(otherwise)

RO

Subsystem Vendor ID Register (Location 2Ch) — WORD

RO

Subsystem ID Register (Location 2Eh) — WORD

RO

Expansion ROM Base Address Register (Location 30h) — DWORD

(for non-VFs and non-empty BAR)

a. Expansion ROM Enable

RW or

RO-Zero

b. Reserved_10-1

RO-Zero

Expansion ROM Base Address Register (Location 30h) — DWORD

(for VFs)

a. Reserved_31-0

RO-Zero

Capabilities Pointer Register (Location 34h) – BYTE

RO

Reserved Register (Location 35h) – 3 BYTES

a. Reserved_23-0

RO-Zero

Reserved Register (Location 38h) – DWORD

a. Reserved_31-0

RO-Zero

13. The following default value checks are performed:

Expansion ROM Base Address Register Default Value (Location 30h) – DWORD

a. Expansion ROM Enable

0

14. For functions under test that have a link, the test is run at each of the following link speeds:

- a. 2.5 GT/s (for devices that are capable). See Section 2.1.2.12 for details.
- b. 5.0 GT/s (for devices that are capable). See Section 2.1.2.13 for details.
- c. 8.0 GT/s (for devices that are capable). See Section 2.1.2.14 for details.

The test *fails* if:

- ☐ For a VF, the Vendor ID field returns other than FFFFh.
- ☐ For a non-VF, the Vendor ID field returns 0000h, or FFFFh, or 0001h.
- ☐ For a Root Complex Event Collector, the Class Code field is not 08 0600h
- ☐ The Subsystem Vendor ID field returns 0000h or FFFFh and Base Class/Sub-Class is not one of the following: 06h/00h-04h; 08h/00h-03h.
- ☐ For a VF, Expansion ROM BAR returns a non-zero value.
- ☐ If implemented, Expansion ROM BAR claims more than 16 MB of space.
- ☐ If implemented, Expansion ROM BAR does not return contiguous 1's across all implemented address bits.
- ☐ If implemented, Expansion ROM BAR does not implement RW bits across all implemented address bits.
- ☐ Any of the register field characteristic tests fail.
- ☐ Any of the default value tests fail.

2.2.32. TD_1_26 Misc Type 1 Config Space Header Registers

The test verifies that if the function under test reports a Type 1 Configuration Space Header, it implements the Vendor ID, Device ID, Revision ID, Class Code, Header Type, I/O Base, I/O Limit, I/O Base Upper 16 Bits, I/O Limit Upper 16 Bits, Memory Base, Memory Limit, Prefetchable Memory Base, Prefetchable Memory Limit, Prefetchable Base Upper 32 Bits, Prefetchable Limit Upper 32 Bits, and Expansion ROM Base Address registers as defined in the relevant specifications. (The following registers are not tested because they control the Configuration Space decoding and may interfere with system operation: Primary Bus Number; Secondary Bus Number; Subordinate Bus Number.)

Relevant Specifications

- ☐ *PCI Express Base Specification*
- ☐ *PCI Local Bus Specification, Revision 3.0*

(See Section 1.3 for additional specification revision details.)

Applicable Device/Port Types

This test is run on all device/port types implementing Type 1 Configuration Space Headers.

Starting Configuration

This test is run with functions starting in the D0-Uninitialized state following the standard initialization procedures in Section 2.1.1. Testing in additional Dx states may be optionally supported. ASPM states of the upstream and downstream ports are disabled for this test.

Overview of Test Steps

Test software performs the following steps:

1. Configure the function under test following the procedure described in Section 2.1.1.
2. Read a byte from location 0Eh (Header Type register). If the value in bits 6-0 is 000 0001b (Type 1 Configuration Space Header) the following checks are performed.
3. Read a WORD from location 00h (Vendor ID register) and perform the following checks:
 - a. The Vendor ID field must return a valid value assigned by the PCI-SIG. (For the purposes of this test program, reads of the Vendor ID field must not return 0000h, FFFFh, or 0001h (CRS Software Visibility notification).)
4. Read 3 bytes from location 09h (Class Code registers) and perform the following checks:
 - a. For a device type of Root Port, Switch Upstream Port, Switch Downstream Port, PCI Express to PCI/PCI-X Bridge or PCI/PCI-X to PCI Express Bridge, the Base Class field (byte at 0Bh) must be 06h and the Sub-Class field (byte at 0Ah) must be 04h.
5. Test software reads and saves (for later restoration) the values in the following registers: I/O Base; I/O Limit; I/O Base Upper 16 Bits; I/O Limit Upper 16 Bits.
6. Test software writes the I/O Space Enable field (Command register) with 1 and then reads it back. If it returns 0, then the fact that this function does not support I/O Space decoding is recorded. Otherwise if the value returns 1 it is recorded that the function supports I/O Space decoding.
7. Test software writes the I/O Space Enable field (Command register) with 0 (to disable I/O Space decoding if supported).
8. If the function does not support I/O Space decoding then the following tests are performed:
 - a. Test software writes a byte of FFh to the I/O Base register (location 1Ch) and then reads back a byte from the same location and if it does not return 00h, the test fails.
 - b. Test software writes a byte of FFh to the I/O Limit register (location 1Dh) and then reads back a byte from the same location and if it does not return 00h, the test fails.
 - c. Test software writes a WORD of FFFFh to the I/O Base Upper 16 Bits register (location 30h) and then reads back a WORD from the same location and if it does not return 0000h, the test fails.
 - d. Test software writes a WORD of FFFFh to the I/O Limit Upper 16 Bits register (location 32h) and then reads back a WORD from the same location and if it does not return 0000h, the test fails.
 - e. Test software reads a DWORD at location 10h (BAR0) and if bit 0 is not 0, then the test fails (if the I/O Space Enable field is RO-Zero, the function cannot claim any I/O Space via BARs).

- f. Test software reads a DWORD at location 14h (BAR1) and if bit 0 is not 0, then the test fails (if the I/O Space Enable field is RO-Zero, the function cannot claim any I/O Space via BARs).
9. If the function supports I/O Space decoding then the following tests are performed:
 - a. Test software reads the byte from the I/O Base register (location 1Ch) and saves this value as [IOV].
 - b. Test software reads the byte from the I/O Limit register (location 1Dh) and compares bits 3-0 of this result with bits 3-0 of [IOV]. If bits 3-0 of the two results are not the same value, the test fails.
 - c. Test software verifies that bits 3-0 of [IOV] contain a valid value (0h or 1h). If an invalid value is returned, the test fails.
 - d. Test software looks at the value of bits 3-0 of [IOV] and if it is 1h, it records that 32 bit I/O Space addressing is supported.
 - e. Test software looks at the value of bits 3-0 of [IOV] and if it is 0h, it writes a byte of FFh to the I/O Base register (location 1Ch) and then reads back a byte from the same location and checks that bits 7-4 are non-zero. If they are non-zero it records that 16 bit I/O Space addressing is supported. Otherwise it reads back the DWORD at location 10h (BAR 0) and if bit 0 is not 1, it then reads back the DWORD at location 14h (BAR1) and if bit 0 is also not 1, the test fails (I/O Space Enable must be RO-Zero if neither I/O BAR nor I/O address ranges are supported).
10. If the function supports 16 bit I/O Space addressing then the following tests are performed:
 - a. Test software writes a byte of FFh to the I/O Base register (location 1Ch) and then reads back a byte from the same location and checks the returned value as follows: bits 7-4 return bits 7-4 of the value written; bits 3-0 return the same value as the previously recorded [IOV]. If any of these are not true, the test fails. Otherwise repeat this step using the following write values: 55h; AAh; CCh; 00h.
 - b. Test software writes a byte of FFh to the I/O Limit register (location 1Dh) and then reads back a byte from the same location and checks the returned value as follows: bits 7-4 return bits 7-4 of the value written; bits 3-0 return the same value as the previously recorded [IOV]. If any of these are not true, the test fails. Otherwise repeat this step using the following write values: 55h; AAh; CCh; 00h.
 - c. Test software writes a WORD of FFFFh to the I/O Base Upper 16 Bits register (location 30h) and then reads back a WORD from the same location and if it does not return 0000h, the test fails.
 - d. Test software writes a WORD of FFFFh to the I/O Limit Upper 16 Bits register (location 32h) and then reads back a WORD from the same location and if it does not return 0000h, the test fails.
11. If the function supports 32 bit I/O Space addressing then the following tests are performed:
 - a. Test software writes a byte of FFh to the I/O Base register (location 1Ch) and then reads back a byte from the same location and checks the returned value as follows: bits 7-4 return bits 7-4 of the value written; bits 3-0 return the same value as the previously recorded [IOV]. If any of these are not true, the test fails. Otherwise repeat this step using the following write values: 55h; AAh; CCh; 00h.

- b. Test software writes a byte of FFh to the I/O Limit register (location 1Dh) and then reads back a byte from the same location and checks the returned value as follows: bits 7-4 return bits 7-4 of the value written; bits 3-0 return the same value as the previously recorded [IOV]. If any of these are not true, the test fails. Otherwise repeat this step using the following write values: 55h; AAh; CCh; 00h.
 - c. Test software writes a WORD of FFFFh to the I/O Base Upper 16 Bits register (location 30h) and then reads back a WORD from the same location and if it does not return the value written, the test fails. Otherwise repeat this step using the following write values: 5555h; AAAAh; CCCCh; 0000h.
 - d. Test software writes a WORD of FFFFh to the I/O Limit Upper 16 Bits register (location 32h) and then reads back a WORD from the same location and if it does not return the value written, the test fails. Otherwise repeat this step using the following write values: 5555h; AAAAh; CCCCh; 0000h.
- 12. Test software restores the previously saved (in step 5) values to the following registers: I/O Base; I/O Limit; I/O Base Upper 16 Bits; I/O Limit Upper 16 Bits.
- 13. Test software reads and saves (for later restoration) the values in the following registers: Memory Base; Memory Limit.
- 14. Test software writes the Memory Space Enable field (Command register) with 0 (to disable Memory Space decoding).
- 15. The following tests are performed:
 - a. Test software writes a WORD of FFFFh to the Memory Base register (location 20h) and then reads back a WORD from the same location and checks the returned value as follows: bits 15-4 return bits 15-4 of the value written; bits 3-0 return 0h. If any of these are not true, the test fails. Otherwise repeat this step using the following write values: 5555h; AAAAh; CCCCh; 0000h.
 - b. Test software writes a WORD of FFFFh to the Memory Limit register (location 22h) and then reads back a WORD from the same location and checks the returned value as follows: bits 15-4 return bits 15-4 of the value written; bits 3-0 return 0h. If any of these are not true, the test fails. Otherwise repeat this step using the following write values: 5555h; AAAAh; CCCCh; 0000h.
- 16. Test software restores the previously saved (in step 13) values to the following registers: Memory Base; Memory Limit.
- 17. Test software reads and saves (for later restoration) the values in the following registers: Prefetchable Memory Base; Prefetchable Memory Limit; Prefetchable Base Upper 32 Bits; Prefetchable Limit Upper 32 Bits.
- 18. Test software writes the Memory Space Enable field (Command register) with 1 and then reads it back. If it returns 0, then the test fails. Otherwise if the value returns 1 it is recorded that the function supports Memory Space decoding.
- 19. Test software writes the Memory Space Enable field (Command register) with 0 (to disable Memory Space decoding).

20. If the function supports Memory Space decoding then the following tests are performed:
 - a. Test software reads the WORD from the Prefetchable Memory Base register (location 24h) and saves this value as [PMEMV].
 - b. Test software reads the WORD from the Prefetchable Memory Limit register (location 26h) and compares bits 3-0 of this result with bits 3-0 of [PMEMV]. If bits 3-0 of the two results are not the same value the test fails.
 - c. Test software verifies that bits 3-0 of [PMEMV] contain a valid value (0h or 1h). If an invalid value is returned, the test fails.
 - d. Test software looks at the value of bits 3-0 of [PMEMV] and if it is 1h, it records that 64 bit Prefetchable Memory Space addressing is supported.
 - e. Test software looks at the value of bits 3-0 of [PMEMV] and if it is 0h, it writes a WORD of FFFFh to the Prefetchable Memory Base register (location 24h) and then reads back the same location and checks that bits 15-4 are non-zero. If they are non-zero it records that 32 bit Prefetchable Memory Space addressing is supported. Otherwise it records that Prefetchable Memory Space addressing is not supported.
21. If the function does not support Prefetchable Memory Space addressing then the following tests are performed:
 - a. Test software writes a WORD of FFFFh to the Prefetchable Memory Base register (location 24h) and then reads back a WORD from the same location and if it does not return 0000h, the test fails.
 - b. Test software writes a WORD of FFFFh to the Prefetchable Memory Limit register (location 26h) and then reads back a WORD from the same location and if it does not return 0000h, the test fails.
 - c. Test software writes a DWORD of FFFF FFFFh to the Prefetchable Base Upper 32 Bits register (location 28h) and then reads back a DWORD from the same location and if it does not return 0000 0000h, the test fails.
 - d. Test software writes a DWORD of FFFF FFFFh to the Prefetchable Limit Upper 32 Bits register (location 2Ch) and then reads back a DWORD from the same location and if it does not return 0000 0000h, the test fails.
22. If the function supports 32 bit Prefetchable Memory Space addressing then the following tests are performed:
 - a. Test software writes a WORD of FFFFh to the Prefetchable Memory Base register (location 24h) and then reads back a WORD from the same location and checks the returned value as follows: bits 15-4 return bits 15-4 of the value written; bits 3-0 return the same value as the previously recorded [PMEMV]. If any of these are not true, the test fails. Otherwise repeat this step using the following write values: 5555h; AAAAh; CCCCh; 0000h.
 - b. Test software writes a WORD of FFFFh to the Prefetchable Memory Limit register (location 26h) and then reads back a WORD from the same location and checks the returned value as follows: bits 15-4 return bits 15-4 of the value written; bits 3-0 return the same value as the previously recorded [PMEMV]. If any of these are not true, the test fails. Otherwise repeat this step using the following write values: 5555h; AAAAh; CCCCh; 0000h.
 - c. Test software writes a DWORD of FFFF FFFFh to the Prefetchable Base Upper 32 Bits register (location 28h) and then reads back a DWORD from the same location and if it does not return 0000 0000h, the test fails.
 - d. Test software writes a DWORD of FFFF FFFFh to the Prefetchable Limit Upper 32 Bits register (location 2Ch) and then reads back a DWORD from the same location and if it does not return 0000 0000h, the test fails.

23. If the function supports 64 bit Prefetchable Memory Space addressing then the following tests are performed:
 - a. Test software writes a WORD of FFFFh to the Prefetchable Memory Base register (location 24h) and then reads back a WORD from the same location and checks the returned value as follows: bits 15-4 return bits 15-4 of the value written; bits 3-0 return the same value as the previously recorded [PMEMV]. If any of these are not true, the test fails. Otherwise repeat this step using the following write values: 5555h; AAAAh; CCCCh; 0000h.
 - b. Test software writes a WORD of FFFFh to the Prefetchable Memory Limit register (location 26h) and then reads back a WORD from the same location and checks the returned value as follows: bits 15-4 return bits 15-4 of the value written; bits 3-0 return the same value as the previously recorded [PMEMV]. If any of these are not true, the test fails. Otherwise repeat this step using the following write values: 5555h; AAAAh; CCCCh; 0000h.
 - c. Test software writes a DWORD of FFFF FFFFh to the Prefetchable Base Upper 32 Bits register (location 28h) and then reads back a DWORD from the same location and if it does not return the value written, the test fails. Otherwise repeat this step using the following write values: 5555 5555h; AAAA AAAAh; CCCC CCCCh; 0000 0000h.
 - d. Test software writes a DWORD of FFFF FFFFh to the Prefetchable Limit Upper 32 Bits register (location 2Ch) and then reads back a DWORD from the same location and if it does not return the value written, the test fails. Otherwise repeat this step using the following write values: 5555 5555h; AAAA AAAAh; CCCC CCCCh; 0000 0000h.
24. Test software restores the previously saved (in step 17) values to the following registers: Prefetchable Memory Base; Prefetchable Memory Limit; Prefetchable Base Upper 32 Bits; Prefetchable Limit Upper 32 Bits.
25. Test software writes FFFF FFFFh to the Expansion ROM Base Address register (location 38h for Type 1 Configuration Space Header) and then reads back a DWORD from the same register and if it is zero the test skips to step 30, but if it is non-zero, the following checks are performed:
 - a. Bit 0 (Expansion ROM Enable field) must return 1.
 - b. Bit 24 must return 1 (ROM BAR must claim 16 MB or less).
 - c. Starting at bit 11 determine the least significant bit value [LSB] that is set to 1. [LSB] must be less than 25. All bits from 31 to [LSB] must also be 1. This same [LSB] value is used in subsequent tests.
26. Test software writes 5555 5555h to the ROM BAR and then reads back a DWORD from the same register. Bits 31-[LSB] must return the written value. Bits [LSB]-11 must return all 0.
27. Test software writes AAAA AAAAh to the ROM BAR and then reads back a DWORD from the same register. Bits 31-[LSB] must return the written value. Bits [LSB]-11 must return all 0.
28. Test software writes CCCC CCCCh to the ROM BAR and then reads back a DWORD from the same register. Bits 31-[LSB] must return the written value. Bits [LSB]-11 must return all 0.
29. Test software writes 0000 0000h to the ROM BAR and then reads back a DWORD from the same register. Bits 31-[LSB] must return the written value. Bits [LSB]-11 must return all 0. Bit 0 (Expansion ROM Enable) must return 0.

30. The following register field characteristic checks are performed:

Vendor ID Register (Location 00h) — WORD	RO
Device ID Register (Location 02h) — WORD	RO
Revision ID Register (Location 08h) — BYTE	RO
Class Code Register (Location 09h) — 3 BYTES	
a. Programming Interface	RO
b. Sub-Class	RO
c. Base Class	RO
Header Type Register (Location 0Eh) — BYTE	RO
Capabilities Pointer Register (Location 34h) – BYTE	RO
Reserved Register (Location 35h) — 3 BYTES	
a. Reserved_23-0	RO-Zero
Expansion ROM Base Address Register (Location 38h) — DWORD (for non-empty BAR)	
a. Expansion ROM Enable	RW or RO-Zero
b. Reserved_10-1	RO-Zero

31. The following default value checks are performed:

Expansion ROM Base Address Register Default Value (Location 38h) — DWORD	
a. Expansion ROM Enable	0

32. For functions under test that have a link, the test is run at each of the following link speeds:

- 2.5 GT/s (for devices that are capable). See Section 2.1.2.12 for details.
- 5.0 GT/s (for devices that are capable). See Section 2.1.2.13 for details.
- 8.0 GT/s (for devices that are capable). See Section 2.1.2.14 for details.

The test *fails* if:

- ☐ The Vendor ID field returns 0000h, FFFFh, or 0001h.
- ☐ For a Root Port, Switch, or Bridge, the Class Code register is not 06 04xxh
- ☐ If I/O Base bits 3-0 does not return the same value as I/O Limit bits 3-0.
- ☐ If I/O Base bits 3-0 returns a reserved value.
- ☐ If I/O Space Enable is RO-Zero, but any of the following occurs: BAR0 bit 0 is not 0; BAR1 bit 0 is not 0; I/O Base is not RO-Zero; I/O Limit is not RO-Zero; I/O Base Upper 16 Bits are not RO-Zero; I/O Limit Upper 16 Bits are not RO-Zero.
- ☐ If 16 bit I/O addressing is reported, but any of the following occurs: I/O Base bits 15-12 are not RW; I/O Base bits 11-4 are not RO-Zero; I/O Base bits 3-0 are not RO; I/O Limit bits 15-12 are not RW; I/O Limit bits 11-4 are not RO-Zero; I/O Limit bits 3-0 are not RO; I/O Base Upper 16 Bits is not RO-Zero; I/O Limit Upper 16 Bits is not RO-Zero.
- ☐ If 32 bit I/O addressing is reported, but any of the following occurs: I/O Base bits 15-12 are not RW; I/O Base bits 11-4 are not RO-Zero; I/O Base bits 3-0 are not RO; I/O Limit bits 15-12 are not RW; I/O Limit bits 11-4 are not RO-Zero; I/O Limit bits 3-0 are not RO; I/O Base Upper 16 Bits is not RW; I/O Limit Upper 16 Bits is not RW.

- ☐ If any of the following occurs: Memory Base bits 15-4 are not RW; Memory Base bits 3-0 are not RO-Zero; Memory Limit bits 15-4 are not RW; Memory Limit bits 3-0 are not RO-Zero
- ☐ If Prefetchable Memory Base bits 3-0 does not return the same value as Prefetchable Memory Limit bits 3-0.
- ☐ If Prefetchable Memory Base bits 3-0 returns a reserved value.
- ☐ If Prefetchable Memory addressing is not reported, but any of the following occurs: Memory Prefetchable Base is not RO-Zero; Memory Prefetchable Limit is not RO-Zero; Prefetchable Base Upper 32 Bits are not RO-Zero; Prefetchable Limit Upper 32 Bits are not RO-Zero.
- ☐ If 32 bit Prefetchable Memory addressing is reported, but any of the following occurs: Memory Prefetchable Base bits 15-4 are not RW; Memory Prefetchable Base bits 3-0 are not RO; Memory Prefetchable Limit bits 15-4 are not RW; Memory Prefetchable Limit bits 3-0 are not RO; Prefetchable Base Upper 32 Bits is not RO-Zero; Prefetchable Limit Upper 32 Bits is not RO-Zero.
- ☐ If 64 bit Prefetchable Memory addressing is reported, but any of the following occurs: Memory Prefetchable Base bits 15-4 are not RW; Memory Prefetchable Base bits 3-0 are not RO; Memory Prefetchable Limit bits 15-4 are not RW; Memory Prefetchable Limit bits 3-0 are not RO; Prefetchable Base Upper 32 Bits is not RW; Prefetchable Limit Upper 32 Bits is not RW.
- ☐ If implemented, Expansion ROM BAR claims more than 16 MB of space.
- ☐ If implemented, Expansion ROM BAR does not return contiguous 1's across all implemented address bits.
- ☐ If implemented, Expansion ROM BAR does not implement RW bits across all implemented address bits.
- ☐ Any of the register field characteristic tests fail.
- ☐ Any of the default value tests fail.

2.2.33. TD_1_27 Multi-Function

The test verifies that if the function under test is part of a multi-function device then all other implemented functions in the same device respond correctly as defined in the relevant specifications.

Relevant Specifications

- ☐ *PCI Express Base Specification*
- ☐ *PCI Local Bus Specification, Revision 3.0*
- ☐ *ECN Alternate Routing-ID Interpretation (to Base 2.0 and Base 1.1)*
- ☐ *ECN Latency Tolerance Reporting (to Base 2.0)*
- ☐ *ECN Multicast (to Base 2.0)*
- ☐ *ECN Optimized Buffer Flush/Fill (to Base 2.0 and Base 2.1)*
- ☐ *Single Root I/O Virtualization and Sharing Specification*

(See Section 1.3 for additional specification revision details.)

Applicable Device/Port Types

This test is run on all device/port types.

Starting Configuration

This test is run with functions starting in the D0-Uninitialized state following the standard initialization procedures in Section 2.1.1. Testing in additional Dx states may be optionally supported. ASPM states of the upstream and downstream ports are disabled for this test.

Overview of Test Steps

Test software performs the following steps:

1. Configure the function under test following the procedure described in Section 2.1.1.
2. Since this test checks the function under test as well as any other functions (both ARI and non-ARI), PFs, and VFs within the device under test, the test must ensure that all the necessary initialization of the hierarchy is done so that the device under test's ARI functions and VFs are visible at this point. This requires that the test re-execute the detection algorithm described in Section 1.4.1.
3. Read a byte from location 0Eh (Header Type register) in the Configuration Space. (For this test, the current function is the function under test. Function 0 is function number 0. Other non-ARI functions are the complete range of function number 1 to function number 7. Other ARI functions are in the complete range of function number 1 to function number 255.)
4. For a VF:
 - a. Check that the current function number is not zero.
 - b. For a non-ARI device, check that the device does respond to configuration requests to Function Number = 000b with the same Bus Number and Device Number. (For the purposes of this test, responding means that reading back a WORD at location 00h (Vendor ID) returns any value other than FFFFh or 0001h (CRS Software Visibility notification).)
 - c. For an ARI device, check that the device does respond to configuration requests to Device Number/Function Number = 00h with the same Bus Number. (For the purposes of this test, responding means that reading back a WORD at location 00h (Vendor ID) returns any value other than FFFFh or 0001h (CRS Software Visibility notification).)
 - d. For Base 2.x or later testing, if the function under test reports the Capability Version field (PCI Express Capabilities register) is 2h or greater:
 - i. Read a DWORD at offset 24h (Device Capabilities 2 register) and record the value of bits 19-18 (OBFF Supported). Check function 0 that it returns the same value for bits 19-18 in its Device Capabilities 2 register.
 - e. If the optional ARI extended capability is found in the current function, check that all other functions (including function 0 and all other VFs) implement the ARI extended capability. (For an ARI device, there can be up to 256 functions.)
 - f. If the optional TPH extended capability is found in the current function, check that all other functions (including function 0 and all other VFs) implement the TPH extended capability.

5. For a non-VF, if the Multi-Function bit (Header Type bit 7) returns 0 and it does not implement an ARI extended capability:
 - a. Check that the current function number is zero. (For a non-ARI device, function 0 is located at Function Number = 000b using the current Device Number.)
 - b. Check that the device does not respond to configuration requests to all other Function Numbers with the same Bus Number and Device Number. (For the purposes of this test, not responding means that reading back a WORD at location 00h (Vendor ID) returns FFFFh or 0001h (CRS Software Visibility notification). Note: VFs will return Vendor ID of FFFFh, so they will not be considered as another function by this test.) If the device responds to configuration requests to any other Function Number with the same Bus Number and Device Number, then a warning message is issued, but this is not treated in itself as a failure.
 - c. Check that the function does not implement a VC extended capability with Extended Capability ID of 0009h.
 - d. Check that the function does not implement a MFVC extended capability (Extended Capability ID of 0008h).
6. For a non-VF, if the Multi-Function bit (Header Type bit 7) returns 0 and it does implement an ARI extended capability:
 - a. Check that the current function number is zero. (For an ARI device, function 0 is located at Device Number = 0 0000b and Function Number = 000b.)
 - b. For an ARI device, read a WORD at offset 04h in the ARI Capability structure.
 - c. For an ARI device, if the Next Function Number (ARI Capability register) is zero, check that the function does not implement a VC extended capability (Extended Capability ID of 0009h).
 - d. For an ARI device, if the Next Function Number (ARI Capability register) is zero, check that the function does not implement a MFVC extended capability (Extended Capability ID of 0008h).
7. For Base 2.x or later testing: if the current function's Multi-Function bit returns 1 and the current function's Device/Port Type is one of: 0000b (Endpoint), 0001b (Legacy Endpoint), 0101b (Switch Upstream Port), or 0111b (PCI Express to PCI/PCI-X Bridge).
 - a. Read a DWORD at offset 0Ch (Link Capabilities register) and record the value. Check all implemented functions in that device that they return the same value in their Link Capabilities register for all the following fields:
 - i. Max Link Speed/Supported Link Speeds.
 - ii. Maximum Link Width
 - iii. Active State Power Management (ASPM) Support
 - iv. L0s Exit Latency
 - v. L1 Exit Latency
 - vi. Port Number
 - vii. If the function contains an ARI Extended Capability: Clock Power Management
 - b. For Base 3.x or later testing, if the function under test reports the Capability Version field (PCI Express Capabilities register) is 2h or greater:
 - i. Read a DWORD at offset 2Ch (Link Capabilities 2 register) and record the value. Check all implemented functions in that device that they return the same value in their Link Capabilities 2 register for all the following fields:
 - a. Supported Link Speeds Vector.
 - b. Crosslink Supported

8. If the current function's Multi-Function bit returns 1 and the current function's Device/Port Type is one of: 0000b (Endpoint), 0001b (Legacy Endpoint), 0101b (Switch Upstream Port), or 0111b (PCI Express to PCI/PCI-X Bridge).
 - a. Read a WORD at offset 12h (Link Status register) and record the value. Check all implemented functions in that device that they return the same value in their Link Status register for all the following fields:
 - i. Slot Clock Configuration
9. For Base 3.x or later: if the current function's Multi-Function bit returns 1 and the current function's Device/Port Type is one of: 0000b (Endpoint), 0001b (Legacy Endpoint), 0101b (Switch Upstream Port), or 0111b (PCI Express to PCI/PCI-X Bridge).
 - a. If the function under test reports the Capability Version field (PCI Express Capabilities register) is 2h or greater:
 - i. Read a WORD at offset 32h (Link Status 2 register) and record the value. Check all implemented functions in that device that they return the same value in their Link Status 2 register for all the following fields:
 - a. Current De-emphasis Level
10. For a non-VF, if the Multi-Function bit returns 1 and the current function number is not zero:
 - a. For a non-ARI device, check that the device does respond to configuration requests to Function Number = 000b with the same Bus Number and Device Number. (For the purposes of this test, responding means that reading back a WORD at location 00h (Vendor ID) returns any value other than FFFFh or 0001h (CRS Software Visibility notification).)
 - b. For an ARI device, check that the device does respond to configuration requests to the Device Number/Function Number = 00h with the same Bus Number. (For the purposes of this test, responding means that reading back a WORD at location 00h (Vendor ID) returns any value other than FFFFh or 0001h (CRS Software Visibility notification).)
 - c. For an ARI device, read a WORD at offset 04h in the ARI Capability structure and if the Next Function Number (ARI Capability register) is non-zero, check that the device does respond to configuration requests to the Device Number/Function Number given by the value returned in Next Function Number (ARI Capability register) with the same Bus Number. (For the purposes of this test, responding means that reading back a WORD at location 00h (Vendor ID) returns any value other than FFFFh or 0001h (CRS Software Visibility notification).)
 - d. For a non-ARI device, check that the function at Function Number = 000b with the same Bus Number and Device Number also has the Multi-Function bit return 1.
 - e. For an ARI device, check that the function at Device Number/Function Number = 00h with the same Bus Number also has the Multi-Function bit return 1.
 - f. For Base 2.x or later testing, if the function under test reports the Capability Version field (PCI Express Capabilities register) is 2h or greater:
 - i. Read a DWORD at offset 24h (Device Capabilities 2 register) and record the value of bits 19-18 (OBFF Supported). Check function 0 that it returns the same value for bits 19-18 in its Device Capabilities 2 register.
 - g. Check that the current function does not implement a VC extended capability (Extended Capability ID of 0002h).

- h. Check that the current function does not implement a MFVC extended capability (Extended Capability ID of 0008h).
 - i. If the optional VC extended capability (Extended Capability ID of 0009h) is found in the current function, check that function 0 implements a MFVC extended capability (Extended Capability ID of 0008h). (For an ARI device, function 0 is located at Device Number = 0 0000b and Function Number = 000b. For a non-ARI device, function 0 is located at Function Number = 000b using the current Device Number.)
 - j. If the optional Multicast extended capability is found in the current function, check that all other functions (including function 0 but excluding VFs) implement the Multicast extended capability. (For an ARI device, there can be up to 256 functions.)
 - k. If the optional ARI extended capability is found in the current function, check that all other functions (including function 0 and all VFs) implement the ARI extended capability. (For an ARI device, there can be up to 256 functions.)
 - l. Check that the current function does not implement a LTR extended capability.
11. For a non-VF, if the Multi-Function bit returns 1 and the current function number is zero (for an ARI device, function 0 is located at Device Number = 0 0000b and Function Number = 000b. For a non-ARI device, function 0 is located at Function Number = 000b using the current Device Number):
- a. Check that the device does respond to configuration requests to at least one of Function Number 1 to 7 with the same Bus Number and Device Number. (For the purposes of this test, responding means that reading back a WORD at location 00h (Vendor ID) returns any value other than FFFFh or 0001h (CRS Software Visibility notification). VFs will return Vendor ID = FFFFh, so they will not be considered as another function by this test.)
 - b. For an ARI device, read a WORD at offset 04h in the ARI Capability structure and check that the Next Function Number (ARI Capability register) is non-zero.
 - c. For an ARI device, check that the device does respond to configuration requests to the Device Number/Function Number given by the value returned in Next Function Number (ARI Capability register) with the same Bus Number. (For the purposes of this test, responding means that reading back a WORD at location 00h (Vendor ID) returns any value other than FFFFh or 0001h (CRS Software Visibility notification). VFs will return Vendor ID = FFFFh, so they will not be considered as another function by this test.)
 - d. Check all implemented functions (excluding VFs) in that device that they must have the Multi-Function bit return 1.
 - e. For Base 2.x or later testing, if the function under test reports the Capability Version field (PCI Express Capabilities register) is 2h or greater:
 - i. Read a DWORD at offset 24h (Device Capabilities 2 register) and record the value of bits 19-18 (OBFF Supported). Check all implemented functions that they return the same value for bits 19-18 in all of their Device Capabilities 2 register.
 - f. If the optional VC extended capability (Extended Capability ID of 0002h) is found in function 0, check that function 0 does not implement either a VC extended capability (Extended Capability ID of 0009h) or a MFVC extended capability (Extended Capability ID of 0008h) and that no other function implements a VC extended capability (Extended Capability ID of 0009h). (For an ARI device, function 0 is located at Device Number = 0 0000b and Function Number = 000b. For a non-ARI device, function 0 is located at Function Number = 000b using the current Device Number.)

- g. Check all implemented functions in that device for the Device Serial Number extended capability and if the optional Device Serial Number extended capability is found, check that either only function 0 implements this Device Serial Number extended capability or that all other functions (excluding VFs) that implement the Device Serial Number extended capability, return the same value for the PCI Express Device Serial Number field (64 bits) as function 0 does. (For an ARI device, function 0 is located at Device Number = 0 0000b and Function Number = 000b. For a non-ARI device, function 0 is located at Function Number = 000b using the current Device Number. For an ARI device, there can be up to 256 functions.)
 - h. If the optional Multicast extended capability is found in the current function, check that all other functions (excluding VFs) implement the Multicast extended capability. (For an ARI device, there can be up to 256 functions.)
 - i. If the optional ARI extended capability is found in the current function, check that all other functions (including all VFs) implement the ARI extended capability. (For an ARI device, there can be up to 256 functions.)
12. For functions under test that have a link, the test is run at each of the following link speeds:
- a. 2.5 GT/s (for devices that are capable). See Section 2.1.2.12 for details.
 - b. 5.0 GT/s (for devices that are capable). See Section 2.1.2.13 for details.
 - c. 8.0 GT/s (for devices that are capable). See Section 2.1.2.14 for details.

The test *fails* if:

- ☐ For a non-VF, the Multi-Function bit is 0 and the function number is not zero.
- ☐ For a non-VF, the Multi-Function bit is 0 and the device responds to configuration requests to another function.
- ☐ For a non-VF, the Multi-Function bit is 1 and the device has not implemented function 0 as a function that has the Multi-Function bit set to 1.
- ☐ For a non-VF in a non-ARI device, the Multi-Function bit is 1 and there is not at least one other PCI Express function implemented.
- ☐ For a non-VF in an ARI device, function 0 reports a Next Function Number field value of zero.
- ☐ For a non-VF in an ARI device, a function other than 0 reports a non-zero Next Function Number field value, but at that address there is no implemented function.
- ☐ The OBFF Supported field is not the same in all implemented functions (for Base 2.x or later testing only).
- ☐ For a multi-function device associated with an upstream port, all implemented functions do not report the same value for each of the individual Link Capabilities register fields: Max Link Speed/Supported Link Speeds; Maximum Link Width; Active State Power Management (ASPM) Support; L0s Exit Latency; L1 Exit Latency; Port Number.
- ☐ For a multi-function ARI device associated with an upstream port, all implemented functions do not report the same value for each of the individual Link Capabilities register fields: Clock Power Management.
- ☐ For a multi-function device associated with an upstream port, all implemented functions do not report the same value for each of the individual Link Capabilities 2 register fields: Supported Link Speeds Vector; Crosslink Supported.

- ❑ For a multi-function device associated with an upstream port, all implemented functions do not report the same value for each of the individual Link Status register fields: Slot Clock Configuration.
- ❑ For a multi-function device associated with an upstream port, all implemented functions do not report the same value for each of the individual Link Status 2 register fields: Current De-emphasis Level (for Base 3.x or later testing only).
- ❑ For a non-VF, a single-function device implements either a VC extended capability (Extended Capability ID of 0009h) or a MFVC extended capability (Extended Capability ID of 0008h).
- ❑ For a non-VF, a single-function device implements an ARI extended capability.
- ❑ For a non-VF, a multi-function device implements a VC extended capability (Extended Capability ID of 0002h) in a function other than function 0.
- ❑ For a non-VF, a multi-function device implements a MFVC extended capability (Extended Capability ID of 0008h) in a function other than function 0.
- ❑ For a non-VF, a multi-function device implements a VC extended capability (Extended Capability ID of 0009h) in any function other than function 0, but does not implement a MFVC extended capability (Extended Capability ID of 0008h) in function 0.
- ❑ For a non-VF, a multi-function device implements a VC extended capability (Extended Capability ID of 0009h) in any function (including function 0), when it also implements a VC extended capability (Extended Capability ID of 0002h) in function 0.
- ❑ For a non-VF, a multi-function device implements both a MFVC extended capability (Extended Capability ID of 0008h) and a VC extended capability (Extended Capability ID of 0002h) in function 0.
- ❑ For a non-VF, a VC Capability using Capability ID of 0002h is present,
- ❑ For a non-VF, a Device Serial Number extended capability is present, this capability is not in function 0, or any additional functions with this capability return a different Device Serial Number field value than function 0 does.
- ❑ For a non-VF, a multi-function device that implements a Multicast extended capability in any function, does not implement it in all functions.
- ❑ For a non-VF, a multi-function device that implements an ARI extended capability in any function, does not implement it in all functions.
- ❑ For a non-VF, a multi-function device implements a LTR extended capability in any function other than function 0.

2.2.34. TD_1_28 Vital Product Data Capability Structure

The test verifies that if the function under test reports a VPD Capability structure, it is implemented as defined in the relevant specifications. (Only Read operations of the VPD area are performed. Write operations are not performed as they would cause VPD data to be over-written.)

Relevant Specifications

❑ *PCI Local Bus Specification*

(See Section 1.3 for additional specification revision details.)

Applicable Device/Port Types

This test is run on all device/port types.

Starting Configuration

This test is run with functions starting in the D0-Uninitialized state following the standard initialization procedures in Section 2.1.1. Testing in additional Dx states may be optionally supported. ASPM states of the upstream and downstream ports are disabled for this test.

Overview of Test Steps

Test software performs the following steps:

1. Configure the function under test following the procedure described in Section 2.1.1.
2. Examine the Capability ID field for each of the function's Capabilities. Determine how many instances of the Capability ID of 03h (Vital Product Data Capability) are found.
3. If a Capability ID of 03h is found for a capability, the following checks are performed for each instance of that capability.
4. The Next Capability Pointer field must be 00h or greater than 3Fh and the lower 2 bits of this field must be 00b.
5. The following register field characteristic checks are performed:

Vital Product Data Capability Header (Offset 00h) — WORD

- | | |
|----------------------------|----|
| a. Capability ID | RO |
| b. Next Capability Pointer | RO |

VPD Address Register (Offset 02h) — WORD

- | | |
|----------------|----|
| a. VPD Address | RW |
|----------------|----|

VPD Data Register (Offset 04h) — DWORD

- | | |
|-------------|----|
| a. VPD Data | RW |
|-------------|----|

6. For this test, bits 15-0 of the WORD at offset 02h in the capability are written at the same time. Bits 14-0 are the DWORD aligned address in the VPD storage component of the VPD data (the bottom two bits must be zero). Bit 15 is a flag that tells the system to either read from or write to the supplied address. Bit 15 also tells when the requested operation has completed.

7. Test software writes 0000h to the WORD at offset 02h and then test software monitors bit 15 until it is set by hardware to indicate that the VPD Data register contains valid data. If bit 15 does not return a value of one within 1 second, the test fails. When bit 15 returns one, the DWORD at offset 04h is read. It must contain four bytes of data from the VPD storage component. The following checks are performed on the 4 bytes of returned data:
 - a. Byte 0 must return 82h (Identifier String tag matching criteria set out in Appendix I of the *PCI Local Bus Specification*).
 - b. Bytes 2-1 must contain the length of the data items (length value is in bytes designate as [length]).
 - c. Bytes 3 to [length] must contain the data.
 - d. Test software continues to read data four bytes at a time until Byte [length] has been read.
 - e. Test software checks that each byte returned from byte 3 to [length] (Product Name for this first data item) is a valid ASCII code (0x00-0x7F). If not, the test fails and is terminated, otherwise continue.
8. Test software reads the next byte to determine the next Data Type.

(For these steps, the test software checks each byte. It does this by writing the byte address divided by four and rounded down, to the WORD at offset 02h, and then test software monitors bit 15 until it is set by hardware to indicate that the VPD Data register contains valid data. If bit 15 does not return a value of one within 1 second, the test fails. When bit 15 returns one, the DWORD at offset 04h is read. It contains four bytes of data from the VPD storage component. These four bytes are used as the four bytes to be checked. Once these four bytes have been checked, the process is repeated to read the next four bytes.)

 - a. If the returned value is 78h (End Tag), then the test passes and terminates, otherwise continue.
 - b. Initialize the checksum value [CHK] to 00h. (The value [CHK] is byte value used to calculate the checksum of the VDR-R Tag structure, by adding each byte read. Any carries out of [CHK] are dropped.) Initialize the "RV" byte address value [RVBA] to 00h. (The value [RVBA] is used to determine if the checksum keyword has been detected, and to locate the actual checksum byte.)
 - c. The returned value must be 90h (VPD-R) or 91h (VPD-W). If not, the test fails and is terminated, otherwise record the current Tag type (VPD-R or VDP-W) and continue.
 - d. If the current TAG type is VPD-R, then update [CHK] by adding the returned value to it (the byte read in step 8).
 - e. The next two bytes are read and combined into a WORD (first byte read is lower byte of WORD) that indicate the byte length of this data item ([length]). The [length] must not exceed 7FFFh minus the current byte offset. If the [length] does exceed this value, the test fails and is terminated, otherwise continue.
 - f. If the current TAG type is VPD-R, then update [CHK] by adding each of the two bytes to it (the two bytes read in step e).
 - g. Test software reads the next byte and checks that it (1st letter of field name) is a valid ASCII code (0x00-0x7F). If not, the test fails and is terminated, otherwise continue.
 - h. If the current TAG type is VPD-R, then update [CHK] by adding the returned value to it (the byte read in step g).
 - i. Test software reads the next byte and checks that it (2nd letter of field name) is a valid ASCII code (0x00-0x7F). If not, the test fails and is terminated, otherwise continue.

- j. If the current TAG type is VPD-R, then update [CHK] by adding the returned value to it (the byte read in step i).
 - k. The WORD (1st letter of field name, 2nd letter of field name) is checked that it is one of the following sets:
 - i. If this current Tag type is VPR-R then ASCII (PN, EC, FG, LC, MN, PG, SN, Vx, CP, RV) – where x is any ASCII character between 0 and Z. If RV (checksum) is detected, then this is recorded.
 - ii. If this current Tag type is VPR-W, then ASCII (Vx, Yx, RW) – where x is any ASCII character between 0 and Z. If RW (remaining WORDs) is detected, then this is recorded.
 - iii. Test software reads the next byte and this value indicates the length of the field [field length].
 - iv. If the current TAG type is VPD-R, then update [CHK] by adding the returned value to it (the byte read in step iii).
 - v. Test software continues to read and check data until either [field length] is reached or [length] is reached. If [length] is reached before [field length] is reached, the test fails and terminates. If [field length] is reached before [length] is reached, the current field name that was recorded is cleared and the test repeats steps 8g – 8k to record the new current field name. If the current field name is not “RV” or “RW”, then the read data is checked that it is a valid ASCII code (0x00-0x7F) and if not a valid ASCII code, the test fails and is terminated. If the current TAG type is VPD-R, then update [CHK] by adding the returned value to it (each byte read in this step). If the current TAG type is VPD-R, and the byte read is the first byte of “RV”, then this byte address is recorded as value [RVBA]. If the current TAG type is VPD-R and the value in [RVBA] is non-zero, and if the byte address of the byte just read is equal to [RVBA] + 3, then the byte just read is the checksum byte, so verify that the value in [CHK] is 00h and if not, the test fails is terminated.
 - l. If the current Tag type is VPD-R and both [length] and [field length] are reached, without detecting a field type of “RV” (the value in [RVBA] is still 0), the test fails and is terminated.
9. If the current byte address is greater than 7FFFh (maximum VPD Address size), the test fails and is terminated, otherwise test software repeats step 8 until an End Tag is found.
10. For functions under test that have a link, the test is run at each of the following link speeds:
- a. 2.5 GT/s (for devices that are capable). See Section 2.1.2.12 for details.
 - b. 5.0 GT/s (for devices that are capable). See Section 2.1.2.13 for details.
 - c. 8.0 GT/s (for devices that are capable). See Section 2.1.2.14 for details.

The test *fails* if:

- ☐ Configuration Space cannot be read.
- ☐ A non-zero Next Capability Pointer is less than or equal to 3Fh, or the lower 2 bits are non-zero.
- ☐ The Flag field does not return 1 upon an attempt to read VPD data.
- ☐ The Tag field cannot be read or is not correct.
- ☐ The VPD area does not begin with a valid Product Name.
- ☐ The VPD area contains a VPD-R Tag type, but it does not contain a RV (checksum) field.
- ☐ The VPD area contains a VPD-R Tag type, but the calculated checksum is not 00h.

- ☐ The VPD area contains a VPD-R Tag type, with reserved or invalid field names.
- ☐ The VPD area contains a VPD-W Tag type, with reserved or invalid field names.
- ☐ The VPD area contains field lengths that exceed the size of the Tag length.
- ☐ The End Tag does not exist.
- ☐ Any of the register field characteristic tests fail.

2.2.35. TD_1_32 PCI-X Capability Structure

The test verifies that if the function under test reports a PCI-X Capability structure, it is implemented as defined in the relevant specifications. (Only a Bridge with a PCI-X bus can implement this capability.)

Relevant Specifications

- ☐ *PCI Express to PCI/PCI-X Bridge Specification, Revision 1.0*
- ☐ *PCI-X Addendum to the PCI Local Bus Specification, Revision 2.0a*

(See Section 1.3 for additional specification revision details.)

Applicable Device/Port Types

This test is run on all device/port types.

Starting Configuration

This test is run with functions starting in the D0-Uninitialized state following the standard initialization procedures in Section 2.1.1. Testing in additional Dx states may be optionally supported. ASPM states of the upstream and downstream ports are disabled for this test.

Overview of Test Steps

Test software performs the following steps:

1. Configure the function under test following the procedure described in Section 2.1.1.
2. Examine the Capability ID field for each of the function's Capabilities. Determine how many instances of the Capability ID of 07h (PCI-X Capability) are found. If more than one is found, the test terminates with a failure.
3. If a Capability ID of 07h is found for a capability, the following checks are performed.
4. The PCI-X capability is only valid for a device type of PCI Express to PCI/PCI-X Bridge or PCI/PCI-X to PCI Express Bridge (a device with a PCI-X bus). For other device types, this capability must not be present, or the test terminates with failure.
5. The Next Capability Pointer field must be 00h or greater than 3Fh and the lower 2 bits of this field must be 00b.
6. Test software reads the WORD from offset 02h (PCI-X Secondary Status register) and performs the following checks:
 - a. The PCI-X Capabilities List Item Version field has a valid encoding (00b, 01b, or 10b).
7. Test software reads the DWORD from offset 10h (PCI-X Bridge ECC Control and Status register) and performs the following checks:
 - a. The ECC Error Phase field has a valid encoding (000b to 101b).

8. The following register field characteristic checks are performed:

PCI-X Capability Header (Offset 00h) — WORD

- | | |
|----------------------------|----|
| a. Capability ID | RO |
| b. Next Capability Pointer | RO |

PCI-X Secondary Status Register (Offset 02h) — WORD

- | | |
|---|---------|
| a. 64-Bit Device | RO |
| b. 133 MHz Capable | RO |
| c. Split Completion Discarded | RW1C |
| d. Unexpected Split Completion | RW1C |
| e. Split Completion Overrun | RW1C |
| f. Split Request Delayed | RW1C |
| g. Secondary Bus Mode and Frequency | RO |
| h. Reserved_11-10 | RO-Zero |
| i. PCI-X Capabilities List Item Version | RO |
| j. PCI-X 266 Capable | RO |
| k. PCI-X 533 Capable | RO |

PCI-X Bridge Status Register (Offset 04h) — DWORD

- | | |
|---------------------------------------|----------|
| a. Function Number | RO |
| b. Device Number | RO |
| c. Bus Number | RO |
| d. 64-bit Device | |
| (for PCI Express to PCI/PCI-X Bridge) | RO-Zero |
| (for PCI/PCI-X to PCI Express Bridge) | RO |
| e. 133 MHz Capable | |
| (for PCI Express to PCI/PCI-X Bridge) | RO-Zero |
| (for PCI/PCI-X to PCI Express Bridge) | RO |
| f. Split Completion Discarded | |
| (for PCI Express to PCI/PCI-X Bridge) | RO-Zero |
| (for PCI/PCI-X to PCI Express Bridge) | RW1C |
| g. Unexpected Split Completion | RW1C |
| h. Split Completion Overrun | RW1C |
| i. Split Request Delayed | RW1C |
| j. Reserved_28-22 | RO-Zero |
| k. Device ID Messaging Capable | RO |
| l. PCI-X 266 Capable | |
| (for PCI Express to PCI/PCI-X Bridge) | RO- Zero |
| (for PCI/PCI-X to PCI Express Bridge) | RO |
| m. PCI-X 533 Capable | |
| (for PCI Express to PCI/PCI-X Bridge) | RO-Zero |
| (for PCI/PCI-X to PCI Express Bridge) | RO |

Upstream Split Transaction Control Register (Offset 08h) — DWORD

- | | |
|---------------------------------------|----|
| a. Split Transaction Capacity | RO |
| b. Split Transaction Commitment Limit | RW |

Downstream Split Transaction Control Register (Offset 0Ch) — DWORD

- | | |
|---------------------------------------|----|
| a. Split Transaction Capacity | RO |
| b. Split Transaction Commitment Limit | RW |

PCI-X Bridge ECC Control and Status Register (Offset 10h) — DWORD

- | | |
|---|----------|
| a. Select Secondary ECC registers | RO |
| b. Error Present in Other ECC Register Bank | RO-Zero |
| c. Additional Correctable ECC Error | RW1C |
| d. Additional Uncorrectable ECC Error | RW1C |
| e. ECC Error Phase | RW1C |
| f. ECC Error Corrected | RO |
| g. Syndrome | RO |
| h. Error First (or only) Command | RO |
| i. Error Second Command | RO |
| j. Error Upper Attributes | RO |
| k. ECC Control Update Enable | RO-Zero |
| l. Reserved_29 | RO-Zero |
| m. Disable Single-Bit-Error Correction | RW or RO |
| n. ECC Mode | RW or RO |

PCI-X Bridge ECC First Address (Offset 14h) — DWORD

- | | |
|-----------------------------|----|
| a. First 32 bits of Address | RO |
|-----------------------------|----|

PCI-X Bridge ECC Second Address (Offset 18h) — DWORD

- | | |
|------------------------------|----|
| a. Second 32 bits of Address | RO |
|------------------------------|----|

PCI-X Bridge ECC Attribute (Offset 1Ch) — DWORD

- | | |
|------------------|----|
| a. ECC Attribute | RO |
|------------------|----|

9. The following default value checks are performed:

PCI-X Secondary Status Register Default Value (Offset 02h) — WORD

- | | |
|--------------------------------|---|
| a. Split Completion Discarded | 0 |
| b. Unexpected Split Completion | 0 |
| c. Split Completion Overrun | 0 |
| d. Split Request Delayed | 0 |

PCI-X Bridge Status Register Default Value (Offset 04h) — DWORD

- | | |
|--------------------------------|---|
| a. Split Completion Discarded | 0 |
| b. Unexpected Split Completion | 0 |
| c. Split Completion Overrun | 0 |
| d. Split Request Delayed | 0 |

Upstream Split Transaction Control Register Default Value (Offset 08h) — DWORD

- | | |
|--|--|
| a. Split Transaction Commitment Limit (value returned from Split Transaction Capacity field) | |
|--|--|

Downstream Split Transaction Control Register Default Value (Offset 0Ch) — DWORD

- | | |
|--|--|
| a. Split Transaction Commitment Limit (value returned from Split Transaction Capacity field) | |
|--|--|

PCI-X Bridge ECC Control and Status Register Default Value (Offset 10h) — DWORD

- | | |
|--|------|
| a. Select Secondary ECC registers
(for PCI Express to PCI/PCI-X Bridge) | 1 |
| (for PCI/PCI-X to PCI Express Bridge) | 0 |
| b. Additional Correctable ECC Error | 0 |
| c. Additional Uncorrectable ECC Error | 0 |
| d. ECC Error Phase | 000b |

10. For functions under test that have a link, the test is run at each of the following link speeds:
- 2.5 GT/s (for devices that are capable). See Section 2.1.2.12 for details.
 - 5.0 GT/s (for devices that are capable). See Section 2.1.2.13 for details.
 - 8.0 GT/s (for devices that are capable). See Section 2.1.2.14 for details.

The test *fails* if:

- ☐ Configuration Space cannot be read.
- ☐ For devices other than a PCI Express to PCI/PCI-X Bridge or a PCI/PCI-X to PCI Express Bridge, the PCI-X capability is present.
- ☐ More than one PCI-X capability structure is present.
- ☐ A non-zero Next Capability Pointer field is less than or equal to 3Fh, or the lower 2 bits are non-zero.
- ☐ The PCI-X Capabilities List Item Version field contains a reserved value.
- ☐ The ECC Error Phase field contains a reserved value.
- ☐ Any of the register field characteristic tests fail.
- ☐ Any of the default value tests fail.

2.2.36. TD_1_30 Root Complex Link Declaration Extended Capability Structure

The test verifies that if the function under test reports a PCI Express Root Complex Link Declaration Extended Capability structure, it is implemented as defined in the relevant specifications. (A Root Complex Link Declaration Extended Capability structure can only be implemented in a Root Port, a Root Complex Integrated Endpoint, or an RCRB. An RCRB can only be implemented in a Root Port or a Root Complex Integrated Endpoint and it cannot reside in a function's Extended Configuration space.)

Relevant Specifications

- ☐ *PCI Express Base Specification*

(See Section 1.3 for additional specification revision details.)

Applicable Device/Port Types

This test is run on all device/port types.

This test is run on any RCRB associated with the function under test.

Starting Configuration

This test is run with functions starting in the D0-Uninitialized state following the standard initialization procedures in Section 2.1.1. Testing in additional Dx states may be optionally supported. ASPM states of the upstream and downstream ports are disabled for this test.

Overview of Test Steps

Test software performs the following steps:

1. Configure the function under test following the procedure described in Section 2.1.1.
2. Examine the Extended Capability ID field for each of the detected Extended Capabilities. Determine how many instances of the Extended Capability ID of 0005h (Root Complex Link Declaration Extended Capability) are found. If more than one is found, the test terminates with a failure.
3. If an Extended Capability ID of 0005h is found for an extended capability the following checks are performed on that extended capability structure:
 4. If not an RCRB, the device type must be a Root Port, or a Root Complex Integrated Endpoint. (Root Complex Link Declaration capability structure can only be found in a Root Port, a Root Complex Integrated Endpoint, or an RCRB and an RCRB structure may not be found in Extended Configuration Space of a device).
 5. The Capability Version field must be 1h.
 6. The Next Capability Offset field must be 000h or greater than 0FFh and the lower 2 bits of this field must be 00b.
 7. A DWORD is read from offset 04h (Element Self Description) and the following checks are performed:
 - a. If the device type is a Root Port or a Root Complex Integrated Endpoint, the Element Type field must be 0h (Configuration Space Element). All other encodings are treated as a failure.
 - b. If this is an RCRB, the Element Type field must be 1h (System Egress Port or internal sink) or 2h (Internal Root Complex Link). All other encodings are treated as a failure.
 - c. The Number of Link Entries field must be non-zero.
 - d. The Component ID field must be non-zero.
 8. The Link Entry counter value [LE] is set to 1.
 9. If [LE] is less than or equal to the Number of Link Entries value [NLE] the following testing is done:
 - a. A DWORD is read from offset $(10h + 10h * ([LE] - 1))$ (the Link Entry [LE] register) and if at least one of the Link Valid field or the Associate RCRB Header field returns 1 the following checks are performed, otherwise skip to step 10:
 - i. The Target Component ID field must be non-zero.
 - ii. If the Associate RCRB Header field returns 1, then all the following must be true: the Link Type field must return 0 (link entry points to memory space); the Element Type field (Element Self Description register) must return 0h (Configuration Space Element); the Device/Port Type field (PCI Express Capabilities register) must be 0100b (Root Port) or 1001b (Root Complex Integrated Endpoint).

10. Increment Link Entry counter value [LE] by adding 1 and repeat step 9, until [LE] is greater than [NLE].

11. The following register field characteristic checks are performed:

Root Complex Link Declaration Extended Capability Header (Offset 00h) — DWORD

- | | |
|---------------------------------------|----|
| a. PCI Express Extended Capability ID | RO |
| b. Capability Version | RO |
| c. Next Capability Offset | RO |

Element Self Description Register (Offset 04h) — DWORD

- | | |
|---------------------------|---------|
| a. Element Type | RO |
| b. RsvdP_7-4 | RO-Zero |
| c. Number of Link Entries | |
| (for non-FLR testing) | HwInit |
| (for FLR testing) | RO |
| d. Component ID | |
| (for non-FLR testing) | HwInit |
| (for FLR testing) | RO |
| e. Port Number | |
| (for non-FLR testing) | HwInit |
| (for FLR testing) | RO |

Reserved Register (Offset 08h) — DWORD

- | | |
|------------------|---------|
| a. Reserved_31-0 | RO-Zero |
|------------------|---------|

Reserved Register (Offset 0Ch) — DWORD

- | | |
|------------------|---------|
| a. Reserved_31-0 | RO-Zero |
|------------------|---------|

12. For each Link Entry register set value [n] (given by value in Number of Link Entries [NLE]), the following register field characteristic checks and register field default value checks are performed:

Link Description Register in Link Entry (n) (Offset 10h + 10h * ([LE] - 1) — WORD

- | | |
|--------------------------|---------|
| a. Link Valid | |
| (for non-FLR testing) | HwInit |
| (for FLR testing) | RO |
| b. Link Type | |
| (for non-FLR testing) | HwInit |
| (for FLR testing) | RO |
| c. Associate RCRB Header | |
| (for non-FLR testing) | HwInit |
| (for FLR testing) | RO |
| d. RsvdP_15-3 | RO-Zero |
| e. Target Component ID | |
| (for non-FLR testing) | HwInit |
| (for FLR testing) | RO |
| f. Target Port Number | |
| (for non-FLR testing) | HwInit |
| (for FLR testing) | RO |

Reserved Register in Link Entry (n) (Offset 14h + 10h * ([LE] -1) — DWORD

- a. Reserved_31-0 RO-Zero

Link Address Register in Link Entry (n) (Offset 18h + 10h * ([LE] -1) — 2 DWORDS

- a. N
 (if Link Type is 1, for non-FLR testing) HwInit
 (if Link Type is 1, for FLR testing) RO
 (if Link Type is 0) RO-Zero
- b. RsvdP_11-3 RO-Zero
- c. Link Address Lower (bits 31-12)
 (for non-FLR testing) HwInit
 (for FLR testing) RO
- d. Link Address Upper (bits 63-32)
 (for non-FLR testing) HwInit
 (for FLR testing) RO

13. For functions under test that have a link, the test is run at each of the following link speeds:

- a. 2.5 GT/s (for devices that are capable). See Section 2.1.2.12 for details.
 b. 5.0 GT/s (for devices that are capable). See Section 2.1.2.13 for details.
 c. 8.0 GT/s (for devices that are capable). See Section 2.1.2.14 for details.

14. If the device type is Root Port or Root Complex Integrated Endpoint, then the test is repeated using each RCRB associated with the function under test, or associated with an RCRB that is itself associated with the function under test. See Section 2.1.2.17 for details. All offsets in the tests are now relative to the RCRB's base address and all cycles are now memory space accesses.

The test *fails* if:

- ☐ More than one Root Complex Link Declaration capability structure is present.
- ☐ The Root Complex Link Declaration capability structure is present in any function other than a Root Port or a Root Complex Integrated Endpoint.
- ☐ The Capability Version field does not report 1h.
- ☐ The Next Capability Offset field does not report 000h or a value greater than 0FFh or the lower two bits are not 00b.
- ☐ The Element Type field is not one of the defined values.
- ☐ The Number of Link Entries field is zero.
- ☐ The Component ID field is zero.
- ☐ The Port Number field is zero, but the Element Type field is not 1h.
- ☐ For any Link Entry with Link Valid field or Associate RCRB field reporting 1, the Target Component ID field is zero.
- ☐ For any Link Entry with Associate RCRB field reporting 1, any of the following occur: the Link Type field is 1; the Element Type field is not 0h; device type is not Root Port or Root Complex Integrated Endpoint.
- ☐ Any of the register field characteristic tests fail.

2.2.37. TD_1_31 Root Complex Internal Link Control Extended Capability Structure

The test verifies that if the function under test reports a PCI Express Root Complex Internal Link Control Extended Capability structure, it is implemented as defined in the relevant specifications. (A Root Complex Internal Link Control Extended Capability structure can only be implemented in an RCRB. An RCRB can only be implemented in a Root Port or a Root Complex Integrated Endpoint and it cannot reside in a function's Extended Configuration space.)

Relevant Specifications

□ *PCI Express Base Specification*

(See Section 1.3 for additional specification revision details.)

Applicable Device/Port Types

This test is run on all device/port types.

This test is run on any RCRB associated with the function under test.

Starting Configuration

This test is run with functions starting in the D0-Uninitialized state following the standard initialization procedures in Section 2.1.1. Testing in additional Dx states may be optionally supported. ASPM states of the upstream and downstream ports are disabled for this test.

Overview of Test Steps

Test software performs the following steps:

1. Configure the function under test following the procedure described in Section 2.1.1.
2. Examine the Extended Capability ID field for each of the detected Extended Capabilities. Determine how many instances of the Extended Capability ID of 0006h (Root Complex Internal Link Control Extended Capability) are found.
3. If the Extended Capability ID of 0006h is found in Extended Configuration Space, the test terminates with a failure.
4. If the Extended Capability ID of 0006h is found in an RCRB the following checks are performed on that extended capability structure:
5. Examine the Extended Capability ID field for each of the detected Extended Capabilities. Determine how many instances of the Extended Capability ID of 0005h (Root Complex Link Declaration Extended Capability) are found. If none are found, then the test terminates with a failure. Otherwise the following checks are performed:
 - a. In the Root Complex Link Declaration Capability (offset 04h), the Element Type field (Element Self Descriptor register) must be 2h (Internal Root Complex Link).
6. If an Extended Capability ID of 0006h is found for an extended capability the following checks are performed on that extended capability structure:
7. The Capability Version field must be 1h.
8. The Next Capability Offset field must be 000h or greater than 0FFh and the lower 2 bits of this field must be 00b.

9. A DWORD is read from offset 04h (Root Complex Link Capabilities register) and the following checks are performed:
 - a. For Base 1.x testing: the Max Link Speed/Supported Link Speeds field must be 0000b or 0001b. All other encodings are reserved and treated as a failure.
 - b. For Base 2.x testing: the Max Link Speed/Supported Link Speeds field must be 0000b, 0001b, or 0010b. All other encodings are reserved and treated as a failure.
 - c. For Base 3.x testing: the Max Link Speed/Supported Link Speeds field must be 0000b, 0001b, 0010b, or 0011b. All other encodings are reserved and treated as a failure.
 - d. The Maximum Link Width field must be 00 0000b, 00 0001b, 00 0010b, 00 0100b, 00 1000b, 00 1100b, 01 0000b, or 10 0000b. All other encodings are reserved and treated as a failure.
 - e. For Base 3.x testing: the Supported Link Speeds Vector field must be 000 0000b, 000 0001b, 000 0010b, 000 0011b, 000 0100b, 000 0101b, 000 0110b, or 000 0111b. All other encodings are reserved and treated as a failure.
10. A WORD is read from offset 0Ah (Root Complex Link Status register) and the following checks are performed:
 - a. For Base 1.x testing: the Current Link Speed field must be 0000b or 0001b. All other encodings are reserved and treated as a failure.
 - b. For Base 2.x testing: the Current Link Speed field must be 0000b, 0001b or 0010b. All other encodings are reserved and treated as a failure.
 - c. For Base 3.x testing: the Current Link Speed field must be 0000b, 0001b, 0010b, or 0011b. All other encodings are reserved and treated as a failure.
 - d. The Current Link Speed field must be equal or less than The Max Link Speed/Supported Link Speeds field (Root Complex Link Capabilities register).
 - e. The Negotiated Link Width field must be 00 0000b, 00 0001b, 00 0010b, 00 0100b, 00 1000b, 00 1100b, 01 0000b, or 10 0000b. All other encodings are reserved and treated as a failure.
 - f. The Negotiated Link Width field must be equal or less than the Maximum Link Width field (Root Complex Link Capabilities register).
11. The following register field characteristic checks are performed:

Root Complex Internal Link Control Extended Capability Header (Offset 00h) — DWORD

- | | |
|---------------------------------------|----|
| a. PCI Express Extended Capability ID | RO |
| b. Capability Version | RO |
| c. Next Capability Offset | RO |

Root Complex Link Capabilities Register (Offset 04h) — DWORD

- | | |
|--|---------|
| a. Max Link Speed/Supported Link Speeds | RO |
| b. Maximum Link Width | RO |
| c. Active State Power Management (ASPM) Support | RO |
| d. L0s Exit Latency | RO |
| e. L1 Exit Latency | RO |
| f. Supported Link Speeds Vector
(For Base 3.x or later testing) | RO |
| g. RsvdP_31-25
(For Base 3.x or later testing) | RO-Zero |

- h. RsvdP_31-18
(For Base 1.x or Base 2.x testing) RO-Zero

Root Complex Link Control Register (Offset 08h) — WORD

- a. Active State Power Management (ASPM) Control
(if Active State Power Management (ASPM) Support is non-zero) RW
(if Active State Power Management (ASPM) Support is 00b) RO-Zero
- b. RsvdP_6-2 RO-Zero
- c. Extended Sync RW or RO-Zero
- d. RsvdP_15-8 RO-Zero

Root Complex Link Status Register (Offset 0Ah) — WORD

- a. Current Link Speed RO
- b. Negotiated Link Width RO
- c. RsvdZ_15-10 RO-Zero

12. The following default value checks are performed:

Root Complex Link Control Register Default Value (Offset 08h) — WORD

- a. Extended Sync 0

13. For functions under test that have a link, the test is run at each of the following link speeds:

- a. 2.5 GT/s (for devices that are capable). See Section 2.1.2.12 for details.
- b. 5.0 GT/s (for devices that are capable). See Section 2.1.2.13 for details.
- c. 8.0 GT/s (for devices that are capable). See Section 2.1.2.14 for details.

14. If the device type is Root Port or Root Complex Integrated Endpoint, then the test is repeated (excluding step 3) using each RCRB associated with the function under test, or associated with an RCRB that is itself associated with the function under test. See Section 2.1.2.17 for details. All offsets in the tests are now relative to the RCRB's base address and all cycles are now memory space accesses.

The test *fails* if:

- ☐ More than one Root Complex Internal Link Control capability structure is present.
- ☐ The Root Complex Internal Link Control capability structure is present in Extended Configuration Space.
- ☐ The Capability Version field does not report 1h.
- ☐ The Next Capability Offset field does not report 000h or a value greater than 0FFh or the lower two bits are not 00b.
- ☐ The Current Link Speed field is not equal or less than the reported the Max Link Speed/Supported Link Speeds field.
- ☐ The Current Link Speed field or the Max Link Speed/Supported Link Speeds field is not one of the defined values.
- ☐ The Negotiated Link Width field is not equal or less than the reported the Maximum Link Width field.
- ☐ The Negotiated Link Width field or the Maximum Link Width field is not one of the defined values.

- ☐ The Supported Link Speeds Vector field is not one of the allowed values.
- ☐ Any of the register field characteristic tests fail.
- ☐ Any of the default value tests fail.

2.2.38. TD_1_34 RCRB Header Extended Capability Structure

The test verifies that if the function under test reports a PCI Express RCRB Header Extended Capability structure, it is implemented as defined in the relevant specifications. (An RCRB Header can only be implemented in Root Port or a Root Complex Integrated Endpoint and it cannot reside in a function's Extended Configuration space.)

Relevant Specifications

- ☐ *PCI Express Base Specification*

(See Section 1.3 for additional specification revision details.)

Applicable Device/Port Types

This test is run on all device/port types.

This test is run on any RCRB associated with the function under test.

Starting Configuration

This test is run with functions starting in the D0-Uninitialized state following the standard initialization procedures in Section 2.1.1. Testing in additional Dx states may be optionally supported. ASPM states of the upstream and downstream ports are disabled for this test.

Overview of Test Steps

Test software performs the following steps:

1. Configure the function under test following the procedure described in Section 2.1.1.
2. Examine the Extended Capability ID field for each of the detected Extended Capabilities. Determine how many instances of the Extended Capability ID of 000Ah (RCRB Header Extended Capability) are found. If more than one is found, the test terminates with a failure.
3. If the Extended Capability ID of 000Ah is found in Extended Configuration Space, the test terminates with a failure.
4. If an Extended Capability ID of 000Ah is found in an RCRB the following checks are performed on that extended capability structure:
5. The Capability Version field must be 1h.
6. The Next Capability Offset field must be 000h or greater than 0FFh and the lower 2 bits of this field must be 00b.
7. A WORD is read from offset 04h (Vendor ID) and the following checks are performed:
 - a. The Vendor ID register must return a valid value assigned by the PCI-SIG. (For the purposes of this test program, reads of the Vendor ID register must not return 0000h, FFFFh, or 0001h (CRS Software Visibility notification).

8. The following register field characteristic checks are performed:

RCRB Header Extended Capability Header (Offset 00h) — DWORD

- | | |
|---------------------------------------|----|
| a. PCI Express Extended Capability ID | RO |
| b. Capability Version | RO |
| c. Next Capability Offset | RO |

Vendor ID Register (Offset 04h) — WORD

RO

Device ID Register (Offset 064h) — WORD

RO

RCRB Capabilities Register (Offset 08h) — DWORD

- | | |
|----------------------------|---------|
| a. CRS Software Visibility | RO |
| b. RsvdP_31-1 | RO-Zero |

RCRB Control Register (Offset 0Ch) — DWORD

- | | |
|---|--------------------------|
| a. CRS Software Visibility Enable
(if CRS Software Visibility is 1)
(if CRS Software Visibility is 0) | RW
RO-Zero
RO-Zero |
| b. RsvdP_31-1 | RO-Zero |

Reserved Register (Offset 10h) — DWORD

- | | |
|---------------|---------|
| a. RsvdZ_31-0 | RO-Zero |
|---------------|---------|

9. The following default value checks are performed:

RCRB Control Register Default Value (Offset 0Ch) — DWORD

- | | |
|-----------------------------------|---|
| a. CRS Software Visibility Enable | 0 |
|-----------------------------------|---|

10. For functions under test that have a link, the test is run at each of the following link speeds:

- 2.5 GT/s (for devices that are capable). See Section 2.1.2.12 for details.
- 5.0 GT/s (for devices that are capable). See Section 2.1.2.13 for details.
- 8.0 GT/s (for devices that are capable). See Section 2.1.2.14 for details.

11. If the device type is Root Port or Root Complex Integrated Endpoint, then the test is repeated using each RCRB associated with the function under test, or associated with an RCRB that is itself associated with the function under test. See Section 2.1.2.17 for details. All offsets in the tests are now relative to the RCRB's base address and all cycles are now memory space accesses.

The test *fails* if:

- ☐ More than one RCRB Header capability structure is present.
- ☐ The RCRB Header capability structure is present in Extended Configuration space.
- ☐ The Capability Version field does not report 1h.
- ☐ The Next Capability Offset field does not report 000h or a value greater than 0FFh or the lower two bits are not 00b.
- ☐ The Vendor ID field returns 0000h, FFFFh, or 0001h.
- ☐ Any of the register field characteristic tests fail.
- ☐ Any of the default value tests fail.

2.2.39. TD_1_29 Root Complex Event Collector Endpoint Association Extended Capability Structure

The test verifies that if the function under test reports a PCI Express Root Complex Event Collector Endpoint Association Extended Capability structure, it is implemented as defined in the relevant specifications. (A Root Complex Link Declaration capability structure can only be implemented in a Root Complex Event Collector.)

Relevant Specifications

□ *PCI Express Base Specification*

(See Section 1.3 for additional specification revision details.)

Applicable Device/Port Types

This test is run on all device/port types.

This test is run on any RCRB associated with the function under test.

Starting Configuration

This test is run with functions starting in the D0-Uninitialized state following the standard initialization procedures in Section 2.1.1. Testing in additional Dx states may be optionally supported. ASPM states of the upstream and downstream ports are disabled for this test.

Overview of Test Steps

Test software performs the following steps:

1. Configure the function under test following the procedure described in Section 2.1.1.
2. Examine the Extended Capability ID field for each of the detected Extended Capabilities. Determine how many instances of the Extended Capability ID of 0007h (Root Complex Event Collector Endpoint Association Extended Capability) are found. If more than one is found, the test terminates with a failure.
3. If the device type is Root Complex Event Collector, then one instance of the Extended Capability ID of 0007h must be found, otherwise the test terminates with a failure.
4. If an Extended Capability ID of 0007h is found for an extended capability the following checks are performed on that extended capability structure:
5. The device type must be Root Complex Event Collector.
6. The Capability Version field must be 1h.
7. The Next Capability Offset field must be 000h or greater than 0FFh and the lower 2 bits of this field must be 00b.

8. A DWORD is read from offset 04h (Association Bitmap for Root Complex Integrated Endpoints) and the following checks are performed:
 - a. The bit position corresponding to the Root Complex Event Collector's Device Number must return 1.
 - b. Starting at bit 0, and proceeding to bit 31, for each bit that returns 1, the following checks are performed:
 - i. If the bit position corresponding to the Root Complex Event Collector's Device Number then this bit position is skipped, and the test continues with the next bit position that returns 1.
 - ii. Test software reads location 00h of the Device Number corresponding to the bit position (Bus Number is the same as the Root Complex Event Collector's Bus Number, and Function Number is 00h) and checks that it returns a Vendor ID field that is not FFFFh.
 - iii. Test software checks for a PCI Express Capability structure for the Device Number corresponding to the bit position, and if it finds one it checks that the device type is Root Complex Integrated Endpoint.
 - iv. The test repeats step b), until all bit positions have been checked.

9. The following register field characteristic checks are performed:

RC Event Collector Endpoint Association Extended Capability Header (Offset 00h) — DWORD

- | | |
|---------------------------------------|----|
| a. PCI Express Extended Capability ID | RO |
| b. Capability Version | RO |
| c. Next Capability Offset | RO |

Association Bitmap for RC Integrated Endpoints Register (Offset 04h) — DWORD
(all bits) RO

10. For functions under test that have a link, the test is run at each of the following link speeds:
 - a. 2.5 GT/s (for devices that are capable). See Section 2.1.2.12 for details.
 - b. 5.0 GT/s (for devices that are capable). See Section 2.1.2.13 for details.
 - c. 8.0 GT/s (for devices that are capable). See Section 2.1.2.14 for details.
11. If the device type is Root Port or Root Complex Integrated Endpoint, then the test is repeated using each RCRB associated with the function under test, or associated with an RCRB that is itself associated with the function under test. See Section 2.1.2.17 for details. All offsets in the tests are now relative to the RCRB's base address and all cycles are now memory space accesses.

The test *fails* if:

- ☐ More than one Root Complex Event Collector Endpoint Association capability structure is present.
- ☐ An RCRB contains a Root Complex Event Collector Endpoint Association capability structure.
- ☐ The Root Complex Event Collector Endpoint Association capability structure is present in any function with a device type other than Root Complex Event Collector.
- ☐ A Root Complex Event Collector does not implement a Root Complex Event Collector Endpoint Association capability structure.
- ☐ The Capability Version field does not report 1h.

- ☐ The Next Capability Offset field does not report 000h or a value greater than 0FFh or the lower two bits are not 00b.
- ☐ The Association Bitmap for Root Complex Endpoints register has the bit position corresponding to the Root Complex Event Collector's Device Number return 0.
- ☐ The Association Bitmap for Root Complex Endpoints register has the bit position corresponding to Device Number that is not a Root Complex Integrated Endpoint return 1.
- ☐ Any of the register field characteristic tests fail.

2.2.40. TD_1_35 Configuration Access Correlation Extended Capability Structure

The test verifies that a Base 2.x or later function under test does not implement the deprecated PCI Express Configuration Access Correlation Extended Capability.

Note: Section B.3 in Appendix B contains the previous test description for this test, as used by previous versions of the test procedure document.

Relevant Specifications

- ☐ *PCI Express Base Specification*
- ☐ *ECN Trusted Configuration Space (to Base 1.1)*

(See Section 1.3 for additional specification revision details.)

Applicable Device/Port Types

This test is run on all device/port types.

This test is run on any RCRB associated with the function under test.

Starting Configuration

This test is run with functions starting in the D0-Uninitialized state following the standard initialization procedures in Section 2.1.1. Testing in additional Dx states may be optionally supported. ASPM states of the upstream and downstream ports are disabled for this test.

Overview of Test Steps

Test software performs the following steps:

1. Configure the function under test following the procedure described in Section 2.1.1.
2. Examine the Extended Capability ID field for each of the detected Extended Capabilities. Determine how many instances of the Extended Capability ID of 000Ch (Configuration Access Correlation Extended Capability) are found. For Base 2.x or later testing: if one or more is found, the test terminates with a failure. Otherwise the test passes.
3. If the device type is Root Port or Root Complex Integrated Endpoint, then the test is repeated using each RCRB associated with the function under test, or associated with an RCRB that is itself associated with the function under test. See Section 2.1.2.17 for details. All offsets in the tests are now relative to the RCRB's base address and all cycles are now memory space accesses.

The test *fails* if:

- ☐ One or more Configuration Access Correlation capability structures are present (for Base 2.x or later testing only).

2.2.41. TD_1_39 Function Level Reset

The test verifies that if a function reports support for Function Level Reset (FLR), the function under test, behaves as defined in the relevant specifications and if it is part of a multi-function device, it is not affected by Function Level Resets in the other functions in the device.

Relevant Specifications

- ❑ *PCI Express Base Specification*
- ❑ *ECN Function Level Reset (to Base 1.1)*
- ❑ *Single Root I/O Virtualization and Sharing Specification*

(See Section 1.3 for additional specification revision details.)

Applicable Device/Port Types

This test is run on all device/port types with functions that support Function Level Reset and are part of a multi-function device.

Starting Configuration

This test is run with functions starting in the D0-Uninitialized state following the standard initialization procedures in Section 2.1.1. Testing in additional Dx states may be optionally supported. ASPM states of the upstream and downstream ports are disabled for this test.

Overview of Test Steps

Test software performs the following steps:

1. Configure the function under test following the procedure described in Section 2.1.1.
2. Since this test uses the function under test as well as any other functions (both ARI and non-ARI), PFs, and VFs within the device under test, the test must ensure that all the necessary initialization of the hierarchy is done so that the device under test's ARI functions and VFs are visible at this point. This requires that the test re-execute the detection algorithm described in Section 1.4.1.
3. Read a DWORD located at offset 04h (Device Capabilities register) in the PCI Express Capability Structure. If bit 28 (Function Level Reset Capability field) is 0, the test is skipped (this is not a failure).
4. Examine the Extended Capability ID field for each of the detected Extended Capabilities. Determine how many instances of the Extended Capability ID of 000Eh (ARI Extended Capability) are found. If one is found, this function is part of an ARI device.
5. Examine the Extended Capability ID field for each of the detected Extended Capabilities. Determine how many instances of the Extended Capability ID of 0010h (SR-IOV Extended Capability) are found. If one is found, this function is a PF and may contain VFs.
 - a. Read the WORD at offset 0Eh (TotalVFs register) in the SR-IOV Extended Capability and if it is non-zero, then this PF contains associated VFs.
6. Read a byte from location 0Eh (Header Type register) in the Configuration Space. If the Multi-Function bit (Header Type register bit 7) returns 1, this function is part of a multi-function device.

7. For this test, the current function is the function under test. Function 0 is Function Number = 000b for a non-ARI device. Function 0 is Device Number/Function Number = 00h for an ARI device. For non-ARI functions, other functions are the complete range of Function Number 000b to 111b (excluding the current function) that return a Vendor ID field that is not FFFFh. For ARI functions, other functions are the complete range of Device Number/Function Number 00h to FFh (excluding the current function) that return a Vendor ID field that is not FFFFh. For PFs with VFs, other functions also include all VFs associated with that PF.
8. If the current function is not a VF, read a byte located at offset 04h (Device Capabilities register) in the PCI Express Capability Structure and take the value in bits 2-0 (Max_Payload_Size Supported field) and write it to offset 08h (Device Control register) bits 7-5 (Max_Payload_Size field).
9. If the current function is not a VF, read a WORD from offset 10h (Link Control register) and then re-write this WORD with all the following fields set to all 1's: Read Completion Boundary (RCB); Common Clock Configuration; Extended Synch; Enable Clock Power Management; Hardware Autonomous Width Disable. (Note: Active State Power Management (ASPM) Control field is not tested as it may affect the function's ASPM state.)
10. Modify some non-sticky field in the current function to a non-default value. (Note: Any of the following fields can be used as long as they are implemented as RW in the target: Memory Space Enable (Command register); I/O Space Enable (Command register), Bus Master Enable (Command register), Interrupt Disable (Command register). If none of these fields are implemented as RW in the target, then other non-sticky RW fields can be used. If the test software cannot find any non-sticky RW field in the target, then this step is skipped.)
11. For the current function, if the Capability Version field (PCI Express Capabilities register) is 2h or greater, then read a WORD from offset 30h (Link Control 2 register) and then re-write this WORD with all the following fields set to all 1's: Hardware Autonomous Speed Disable.
12. If the current function supports either a VC Extended Capability (Extended Capability ID of 0002h or 0009h) or a MFVC Extended Capability (Extended Capability ID of 0008h), then for each implemented VC, test software writes the current function's TC/VC Map field so that only the TC bit matching the VC number is 1 and the other TCs are 0 (for VC0: TC0=1, TC1-7=0; for VC7: TC0-6=0, TC7=1).
13. For the current function only, read the following register fields and record their current value:

Device Capabilities Register (Offset 04h) – DWORD

- a. Captured Slot Power Limit Value
- b. Captured Slot Power Limit Scale

Device Control Register (Offset 08h) – WORD

- a. Max_Payload_Size

Link Control Register (Offset 10h) – WORD

- a. Active State Power Management (ASPM) Control
- b. Read Completion Boundary (RCB)
- c. Common Clock Configuration
- d. Extended Synch
- e. Enable Clock Power Management
- f. Hardware Autonomous Width Disable

Link Control 2 Register (Offset 30h) – WORD (if PCI Express Capability Version > 1)

- a. Hardware Autonomous Speed Disable

Virtual Channel Capability Structure (if present)

- a. TC/VC Map field in all VC Resource Control registers

Multi Function Virtual Channel Capability Structure (if present)

- a. TC/VC Map field in all VC Resource Control registers

Secondary PCI Express Capability Structure (if present)

- a. All Equalization Control registers

14. For the current function, read a byte from location 0Eh (Header Type register) in the Configuration Space. If the Multi-Function bit (Header Type register bit 7) is not set, skip to step 18.
15. For each other function in the device, read a DWORD located at offset 04h (Device Capabilities register) in the PCI Express Capability Structure. If bit 28 (Function Level Reset Capability field) is 1, write a WORD to offset 08h (Device Control register) with bit 15 (Initiate Function Level Reset field) set to 1. After writing each individual function, test software waits 100 ms before writing the next function. After the last function is written, test software waits 100 ms before proceeding.
16. If a non-sticky RW field was found in step 10, test software verifies the same non-sticky field (from step 10) in the current function only is not reset to its default value. If the value has changed, the test fails.
17. For the current function only, read all the register fields listed in step 13 and check that their values are the same as the previously record value. If any value has changed, the test fails.
18. For the current function, read a DWORD located at offset 04h (Device Capabilities register) in the PCI Express Capability Structure. If bit 28 (Function Level Reset Capability field) is 1, write a WORD to offset 08h (Device Control register) with bit 15 (Initiate Function Level Reset field) set to 1. After writing, test software waits 100 ms before proceeding.
19. If a non-sticky RW field was found in step 10, test software verifies the same non-sticky field (from step 10) in the current function only is reset to its default value. If the value has not changed to the default value, the test fails.
20. For the current function only, read all the register fields listed in step 13 and check that their values are the same as the previously record value. If any value has changed, the test fails.
21. Since a FLR to a PF will disable all VFs, the test must ensure that all the necessary initialization of the hierarchy is done so that the device under test's ARI functions and VFs are visible again at this point. This requires that the test re-execute the detection algorithm described in Section 1.4.1.
22. For functions under test that have a link, the test is run at each of the following link speeds:
 - a. 2.5 GT/s (for devices that are capable). See Section 2.1.2.12 for details.
 - b. 5.0 GT/s (for devices that are capable). See Section 2.1.2.13 for details.
 - c. 8.0 GT/s (for devices that are capable). See Section 2.1.2.14 for details.

The test *fails* if:

- ☐ Configuration Space registers cannot be read
- ☐ A non-sticky register field value has changed following the Function Level Reset to another function.
- ☐ A non-sticky register field value has not changed following the Function Level Reset to the function.
- ☐ Any of the listed register fields values have changed following the Function Level Reset to another function.
- ☐ Any of the listed register fields values have changed following the Function Level Reset to the function.

2.2.42. TD_1_42 ACS Extended Capability Structure

The test verifies that if the PCI Express function under test reports an ACS Extended Capability structure, it is implemented as defined in the relevant specifications.

Relevant Specifications

- ☐ *PCI Express Base Specification*
- ☐ *ECN Access Control Services (to Base 1.1)*

(See Section 1.3 for additional specification revision details.)

Applicable Device/Port Types

This test is run on all device/port types.

This test is run on any RCRB associated with the function under test.

Starting Configuration

This test is run with functions starting in the D0-Uninitialized state following the standard initialization procedures in Section 2.1.1. Testing in additional Dx states may be optionally supported. ASPM states of the upstream and downstream ports are disabled for this test.

Overview of Test Steps

Test software performs the following steps:

1. Configure the function under test following the procedure described in Section 2.1.1.
2. Examine the Extended Capability ID field for each of the detected Extended Capabilities. Determine how many instances of the Extended Capability ID of 000Dh (ACS Extended Capability) are found. If more than one is found, the test terminates with a failure.
3. If the Extended Capability ID of 000Dh is found in an RCRB, the test terminates with a failure.
4. If an Extended Capability ID of 000Dh is found for an extended capability the following checks are performed on that extended capability structure:
5. The Capability Version field must be 1h.
6. The Next Capability Offset field must be 000h or greater than 0FFh and the lower 2 bits of this field must be 00b.

7. For a non-multi-function device, the device type must be a Root Port or a Switch Downstream Port.
8. For a multi-function device, the device type must be an Endpoint or a Switch Upstream Port.
9. A WORD is read from offset 04h (ACS Capability register).
10. If the device type is a Root Port or a Switch Downstream Port, the ACS Source Validation field must be 1. For other device types it must be 0.
11. If the device type is a Root Port or a Switch Downstream Port, the ACS Translation Blocking field must be 1. For other device types it must be 0.
12. If the device type is a Switch Downstream Port, the ACS P2P Request Redirect field must be 1.
13. If the device type is a Switch Downstream Port, the ACS Upstream Forwarding field must be 1. For Switch Upstream Port and Endpoint device types the field must be 0.
14. If the device type is a Switch Downstream Port, the ACS Direct Translated P2P field must be 1.
15. If the ACS P2P Request Redirect field is 1, then the ACS P2P Completion Redirect field must be 1.
16. A WORD is read from offset 06h (ACS Control register).
17. If the ACS P2P Egress Control field is 1:
 - a. Calculate the Egress Vector Size [EVS] value as follows: If the Egress Control Vector Size field is 00h, then [EVS] = 8. If the Egress Control Vector Size field is not 00h, then [EVS] = (the vector size divided by 32, rounded up to the next integer number).
 - b. If the function under test is part of an ARI device, the [EVS] value must be any power of 2 between 8 and 256.
 - c. Test software reads [EVS] DWORDs starting from offset 08h (Egress Control Vector register). It constructs the Egress Control Vector value by combining all the read DWORDs into a single value such that, the last DWORD read is the most significant DWORD of the vector, down to the first DWORD read which is the least significant DWORD of the vector. The vector can be up to 256 bits long and the least significant bit of the vector represents port/function 0.
 - d. For a Root Port or Switch Downstream Port, the bit in the Egress Control Vector corresponding to the current port number must be RO-Zero. The rest of the Egress Control Vector must be RW for the number of bits indicated in the Egress Control Vector Size (however, if the function under test is a Root Port a bit corresponding to an internal port may also be RO-Zero.). Any remaining bits must be RO-Zero (RsvdP).
 - e. For a function in a non-ARI device, the bit corresponding to the current function number of the function under test must be RO-zero. The rest of the Egress Control Vector must be RW for the number of bits indicated in the Egress Control Vector Size field. Any remaining bits must be RO-Zero (RsvdP).
18. The following register field characteristic checks are performed:

ACS Extended Capability Header (Offset 00h) — DWORD	
a. PCI Express Extended Capability ID	RO
b. Capability Version	RO
c. Next Capability Offset	RO

ACS Capability Register (Offset 04h) — WORD

a. ACS Source Validation	RO
b. ACS Translation Blocking	RO
c. ACS P2P Request Redirect	RO
d. ACS P2P Completion Redirect	RO
e. ACS Upstream Forwarding	RO
f. ACS P2P Egress Control	RO
g. ACS Direct Translated P2P	RO
h. RsvdP_7	RO-Zero
i. Egress Control Vector Size (for non-FLR testing) (for FLR testing)	HwInit RO

ACS Control Register (Offset 06h) — WORD

a. ACS Source Validation Enable (if ACS Source Validation is 0) (if ACS Source Validation is 1)	RO-Zero RW
b. ACS Translation Blocking Enable (if ACS Translation Blocking is 0) (if ACS Translation Blocking is 1)	RO-Zero RW
c. ACS P2P Request Redirect Enable (if ACS P2P Request Redirect is 0) (ACS P2P Request Redirect is 1)	RO-Zero RW
d. ACS P2P Completion Redirect Enable (if ACS P2P Completion Redirect is 0) (if ACS P2P Completion Redirect is 1)	RO-Zero RW
e. ACS Upstream Forwarding Enable (if ACS Upstream Forwarding is 0) (if ACS Upstream Forwarding is 1)	RO-Zero RW
f. ACS P2P Egress Control Enable (if ACS P2P Egress Control is 0) (if ACS P2P Egress Control is 1)	RO-Zero RW
g. ACS Direct Translated P2P Enable (if ACS Direct Translated P2P is 0) (if ACS Direct Translated P2P is 1)	RO-Zero RW
h. RsvdP_15-7	RO-Zero

19. The following default value checks are performed:

ACS Control Register Default Value (Offset 06h) — WORD

a. ACS Source Validation Enable	0
b. ACS Translation Blocking Enable	0
c. ACS P2P Request Redirect Enable	0
d. ACS P2P Completion Redirect Enable	0
e. ACS Upstream Forwarding Enable	0
f. ACS P2P Egress Control Enable	0
g. ACS Direct Translated P2P Enable	0

20. For functions under test that have a link, the test is run at each of the following link speeds:
 - a. 2.5 GT/s (for devices that are capable). See Section 2.1.2.12 for details.
 - b. 5.0 GT/s (for devices that are capable). See Section 2.1.2.13 for details.
 - c. 8.0 GT/s (for devices that are capable). See Section 2.1.2.14 for details.
21. If the device type is Root Port or Root Complex Integrated Endpoint, then the test is repeated using each RCRB associated with the function under test, or associated with an RCRB that is itself associated with the function under test. See Section 2.1.2.17 for details. All offsets in the tests are now relative to the RCRB's base address and all cycles are now memory space accesses.

The test *fails* if:

- ☐ *More than one ACS capability structure is present.*
- ☐ *An RCRB contains an ACS capability structure.*
- ☐ *The ACS capability structure is present in any function other than the following: Root Port; Switch Downstream Port; multi-function Endpoint device; multi-function Switch with Upstream Port.*
- ☐ *The Capability Version field does not report 1b.*
- ☐ *The Next Capability Offset field does not report 000h or a value greater than 0FFh or the lower two bits are not 00b.*
- ☐ *The device is a Root Port and it does not report the following fields as 1: ACS Source Validation; ACS Translation Blocking.*
- ☐ *The device is a Switch Downstream Port and it does not report the following fields as 1: ACS Source Validation; ACS Translation Blocking; ACS P2P Request Redirect; ACS Upstream Forwarding; ACS Direct Translated P2P.*
- ☐ *The device is a Switch Upstream Port or an Endpoint and it reports the following fields as 1: ACS Source Validation; ACS Translation Blocking; ACS Upstream Forwarding.*
- ☐ *The function reports the ACS P2P Request Redirect field as 1, but does not report the ACS P2P Completion Redirect field as 1.*
- ☐ *The function reports the ACS P2P Egress Control field as 1 but the Egress Control Vector is not implemented correctly, either in terms of the vector size, or the vector bit attributes.*
- ☐ *Any of the register field characteristic tests fail.*
- ☐ *Any of the default value tests fail.*

2.2.43. TD_1_43 ARI Extended Capability Structure

The test verifies that if the function under test reports an Alternate Routing ID (ARI) Extended Capability structure, it is implemented as defined in the relevant specifications.

Relevant Specifications

- ☐ *PCI Express Base Specification*
- ☐ *ECN Alternate Routing-ID Interpretation (to Base 2.0 and Base 1.1)*
- ☐ *ECN Access Control Services (to Base 1.1)*
- ☐ *Single Root I/O Virtualization and Sharing Specification*

(See Section 1.3 for additional specification revision details.)

Applicable Device/Port Types

This test is run on all device/port types.

This test is run on any RCRB associated with the function under test.

Starting Configuration

This test is run with functions starting in the D0-Uninitialized state following the standard initialization procedures in Section 2.1.1. Testing in additional Dx states may be optionally supported. ASPM states of the upstream and downstream ports are disabled for this test.

Overview of Test Steps

Test software performs the following steps:

1. Configure the function under test following the procedure described in Section 2.1.1.
2. Examine the Extended Capability ID field for each of the detected Extended Capabilities. Determine how many instances of the Extended Capability ID of 000Eh (ARI Extended Capability) are found. If more than one is found, the test terminates with a failure.
3. If the Extended Capability ID of 000Eh is found in an RCRB, the test terminates with a failure.
4. If an Extended Capability ID of 000Eh is found for an extended capability the following checks are performed on that extended capability structure:
5. The Capability Version field must be 1h.
6. The Next Capability Offset field must be 000h or greater than 0FFh and the lower 2 bits of this field must be 00b.
7. The device type must be an Endpoint or a Switch Upstream Port, (but not a Root Complex Integrated Endpoint).
8. A WORD is read from offset 04h (ARI Capability register).
9. If the MFVC Function Groups Capability field reports 1, then the function number must be 0.
10. If the ACS Function Groups Capability field reports 1, then the function number must be 0.
11. If the Next Function Number field is non-zero, it must be a value greater than the current function number.
12. The following register field characteristic checks are performed:

ARI Extended Capability Header (Offset 00h) — DWORD

- | | |
|---------------------------------------|----|
| a. PCI Express Extended Capability ID | RO |
| b. Capability Version | RO |
| c. Next Capability Offset | RO |

ARI Capability Register (Offset 04h) — WORD

- | | |
|--|---------|
| a. MFVC Function Groups Capability | RO |
| b. ACS Function Groups Capability | RO |
| c. RsvdP_7-2 | RO-Zero |
| d. Next Function Number
(for non-VFs) | RO |

ARI Control Register (Offset 06h) — WORD

- | | |
|--------------------------------|--|
| a. MFVC Function Groups Enable | |
|--------------------------------|--|

- | | |
|---|---------|
| (if MFVC Function Groups Capability is 0 or function number is non-zero) | RO-Zero |
| (if MFVC Function Groups Capability is 1 and function number is zero) | RW |
| b. ACS Function Groups Enable | |
| (if ACS Function Groups Capability is 0 or function number is non-zero) | RO-Zero |
| (if ACS Function Groups Capability is 1 and function number is zero) | RW |
| c. RsvdP_3-2 | RO-Zero |
| d. Function Group | |
| (if Function 0 of the device reports both MFVC Function Groups Capability and ACS Function Groups Capability as 0) | RO-Zero |
| (if Function 0 of the device reports either MFVC Function Groups Capability or ACS Function Groups Capability as 1) | RW |
| e. RsvdP_15-7 | RO-Zero |
13. The following default value checks are performed:
- ARI Control Register Default Value (Offset 06h) — WORD**
- | | |
|--------------------------------|------|
| a. MFVC Function Groups Enable | 0 |
| b. ACS Function Groups Enable | 0 |
| c. Function Group | 000b |
14. For functions under test that have a link, the test is run at each of the following link speeds:
- 2.5 GT/s (for devices that are capable). See Section 2.1.2.12 for details.
 - 5.0 GT/s (for devices that are capable). See Section 2.1.2.13 for details.
 - 8.0 GT/s (for devices that are capable). See Section 2.1.2.14 for details.
15. If the device type is Root Port or Root Complex Integrated Endpoint, then the test is repeated using each RCRB associated with the function under test, or associated with an RCRB that is itself associated with the function under test. See Section 2.1.2.17 for details. All offsets in the tests are now relative to the RCRB's base address and all cycles are now memory space accesses.

The test *fails* if:

- ☐ More than one ARI capability structure is present.
- ☐ An RCRB contains an ARI capability structure.
- ☐ The ARI capability structure is present in any function other than the following: Endpoint; Legacy Endpoint; Switch Upstream Port.
- ☐ The Capability Version field does not report 1h.
- ☐ The Next Capability Offset field does not report 000h or a value greater than 0FFh or the lower two bits are not 00b.
- ☐ A function other than 0, reports the MFVC Function Groups Capability field as 1.
- ☐ A function other than 0, reports the ACS Function Groups Capability field as 1.
- ☐ A function reports a non-zero Next Function Number field that is lower than the function's number.
- ☐ Any of the register field characteristic tests fail.
- ☐ Any of the default value tests fail.

2.2.44. TD_1_44 DPA Extended Capability Structure

The test verifies that if the function under test reports a Dynamic Power Allocation (DPA) Extended Capability structure, it is implemented as defined in the relevant specifications.

Relevant Specifications

- ☐ *PCI Express Base Specification*
- ☐ *ECN Dynamic Power Allocation (to Base 2.0)*

(See Section 1.3 for additional specification revision details.)

Applicable Device/Port Types

This test is run on all device/port types.

This test is run on any RCRB associated with the function under test.

Starting Configuration

This test is run with functions starting in the D0-Uninitialized state following the standard initialization procedures in Section 2.1.1. Testing in additional Dx states may be optionally supported. ASPM states of the upstream and downstream ports are disabled for this test.

Overview of Test Steps

Test software performs the following steps:

1. Configure the function under test following the procedure described in Section 2.1.1.
2. Examine the Extended Capability ID field for each of the detected Extended Capabilities. Determine how many instances of the Extended Capability ID of 0016h (DPA Extended Capability) are found. If more than one is found, the test terminates with a failure.
3. If the Extended Capability ID of 0016h is found in an RCRB, the test terminates with a failure.
4. If the Extended Capability ID of 0016h is found in a VF, the test terminates with a failure.
5. For Base 1.x testing: if the Extended Capability ID of 0016h is found for an extended capability, the test terminates with a failure.
6. For Base 2.x or later testing: if an Extended Capability ID of 0016h is found for an extended capability the following checks are performed on that extended capability structure:
7. The Capability Version field must be 1h.
8. The Next Capability Offset field must be 000h or greater than 0FFh and the lower 2 bits of this field must be 00b.
9. The device type must be an Endpoint.
10. A DWORD is read from offset 04h (DPA Capability register).
11. The Transition Latency Unit field must return a valid value (00b, 01b, or 10b).
12. A DWORD is read from offset 08h (DPA Latency Indicator register).
13. The value returned in the Substate_Max field is used to determine the upper implemented bit. All bits in the DPA Latency Indicator register above the upper implemented bit must return 0.

14. The following register field characteristic checks are performed:

DPA Extended Capability Header (Offset 00h) — DWORD

- | | |
|---------------------------------------|----|
| a. PCI Express Extended Capability ID | RO |
| b. Capability Version | RO |
| c. Next Capability Offset | RO |

DPA Capability Register (Offset 04h) — DWORD

- | | |
|-------------------------------|---------|
| a. Substate_Max | RO |
| b. RsvdP_7-5 | RO-Zero |
| c. Transition Latency Unit | RO |
| d. RsvdP_11-10 | RO-Zero |
| e. Power Allocation Scale | RO |
| f. RsvdP_15-14 | RO-Zero |
| g. Transition Latency Value 0 | RO |
| h. Transition Latency Value 1 | RO |

DPA Latency Indicator Register (Offset 08h) — DWORD

- | | |
|---|---------|
| a. (bits [Substate_Max] to 0) | RO |
| b. (if [Substate_Max] is less than 31: bits 31 to [Substate_Max] + 1) | RO-Zero |

DPA Status Register (Offset 0Ch) — WORD

- | | |
|-----------------------------|---------|
| a. Substate Status | RO |
| b. RsvdZ_7-5 | RO-Zero |
| c. Substate Control Enabled | RW1C |
| d. RsvdZ_15-9 | RO-Zero |

DPA Control Register (Offset 0Eh) — WORD

- | | |
|---------------------|---------|
| a. Substate Control | RW |
| b. RsvdP_15-5 | RO-Zero |

15. The following default value checks are performed:

DPA Status Register Default Value (Offset 0Ch) — WORD

- | | |
|-----------------------------|---------|
| a. Substate Status | 0 0000b |
| b. Substate Control Enabled | 1 |

DPA Control Register Default Value (Offset 0Eh) — WORD

- | | |
|---------------------|---|
| a. Substate Control | 0 |
|---------------------|---|

16. Determine the number of implemented substates and create the value [SSMax] as the value returned in Substate_Max plus 1.

17. Create the value of the previous DPA Power Allocation Array byte and set [PrvDPA] to FFh, in order to set it to the maximum possible value.

18. Read byte starting at offset 10h (DPA Power Allocation Array).

- Record the byte value in [CurDPA], which is the holder of the current DPA Power Allocation Array byte.
- Verify that [CurDPA] is less than or equal to [PrvDPA].
- Verify that [CurDPA] is non-zero. If it is zero, then a warning message is issued, but this is not treated in itself as a failure.
- Write to the byte with an inverted data pattern.
- Read the same byte again.

- f. Verify that the returned value still matches the value in [CurDPA] (this verifies the array is RO).
 - g. Copy [CurDPA] to [PrvDPA], to prepare for the next byte.
19. Repeat step 18 by continuing to read and check subsequent bytes until [SSMax] bytes are read and checked.
20. Test software tests each supported substate can be reached as follows:
 - a. If [SSMax] is greater than 1, then set the desired substate value [SS] to 1. If [SSMax] is 1, set the desired substate value [SS] to 0.
 - b. Read the Substate Status field and record the value (to be called previous substate).
 - c. Calculate the transition latency delay value [delay] as follows for the desired substate [SS]:
 - i. Use [SS] as a bit index into the DPA Latency Indicator register and determine if the corresponding bit returns 0 or 1.
 - ii. If the bit returned 0, then [delay] is the value returned in the Transition Latency Value 0 field multiplied by the factor returned in the Transition Latency Unit field.
 - iii. If the bit returned 1, then [delay] is the value returned in the Transition Latency Value 1 field multiplied by the factor returned in the Transition Latency Unit field.
 - d. Write [SS] to the Substate Control field,
 - e. Delay for a period of time equal to [delay]
 - f. Read the Substate Status field to verify the function's current substate.
 - i. If the Substate Control Enabled field returns 1, then if the Substate Status field does not return the same value as [SS], the test fails, otherwise continue.
 - ii. If the Substate Control Enabled field returns 0, then if the Substate Status field does not return the same value as the previous substate, the test fails, otherwise continue.
 - g. Increment [SS] and repeat steps b-g until [SSMax] is reached.
 - h. If [SSMax] is greater than 1, then set the desired substate value [SS] to 0.
 - i. If [SSMax] is greater than 1, then repeat steps b-f just once more and then terminate.
21. Write the Substate Control Enabled field with 1 (to clear the RW1C field) and repeat step 20 (this verifies that substate control can be disabled).
22. For functions under test that have a link, the test is run at each of the following link speeds:
 - a. 2.5 GT/s (for devices that are capable). See Section 2.1.2.12 for details.
 - b. 5.0 GT/s (for devices that are capable). See Section 2.1.2.13 for details.
 - c. 8.0 GT/s (for devices that are capable). See Section 2.1.2.14 for details.
23. If the device type is Root Port or Root Complex Integrated Endpoint, then the test is repeated using each RCRB associated with the function under test, or associated with an RCRB that is itself associated with the function under test. See Section 2.1.2.17 for details. All offsets in the tests are now relative to the RCRB's base address and all cycles are now memory space accesses.

The test *fails* if:

- ☐ More than one DPA capability structure is present.
- ☐ An RCRB contains a DPA capability structure.
- ☐ A VF contains a DPA capability structure.
- ☐ The DPA capability structure is present in any function other than the following: Endpoint; Legacy Endpoint; Root Complex Integrated Endpoint.

- ☐ The Capability Version field does not report 1h.
- ☐ The Next Capability Offset field does not report 000h or a value greater than 0FFh or the lower two bits are not 00b.
- ☐ The Transition Latency Unit field returns a reserved value.
- ☐ The DPA Transition Latency register returns a non-zero bit for an unsupported substate.
- ☐ Any of the register field characteristic tests fail.
- ☐ Any of the default value tests fail.
- ☐ The DPA Power Allocation Array returns a higher value for a supported substate, then the preceding substate.
- ☐ The function is unable to reach a supported substate, in the latency time reported.
- ☐ Clearing the Substate Control Enable field does not prevent substate transitions from occurring.

2.2.45. TD_1_45 Resizable BAR Extended Capability Structure

The test verifies that if the function under test reports a Resizable BAR Extended Capability structure, it is implemented as defined in the relevant specifications.

Relevant Specifications

- ☐ *PCI Express Base Specification*
- ☐ *ECN Resizable BAR Capability (to Base 2.0)*

(See Section 1.3 for additional specification revision details.)

Applicable Device/Port Types

This test is run on all device/port types.

This test is run on any RCRB associated with the function under test.

Starting Configuration

This test is run with functions starting in the D0-Uninitialized state following the standard initialization procedures in Section 2.1.1. Testing in additional Dx states may be optionally supported. ASPM states of the upstream and downstream ports are disabled for this test.

Overview of Test Steps

Test software performs the following steps:

1. Configure the function under test following the procedure described in Section 2.1.1.
2. Test software writes 0 to both the Memory Space Enable field and the Bus Master Enable field in the Command register (this disables responses to memory cycles targeting the function's BAR spaces and prevents the function from generating any requests).
3. Examine the Extended Capability ID field for each of the detected Extended Capabilities. Determine how many instances of the Extended Capability ID of 0015h (Resizable BAR Extended Capability) are found. If more than one is found, the test terminates with a failure.
4. If the Extended Capability ID of 0015h is found in an RCRB, the test terminates with a failure.

5. For Base 1.x testing: if the Extended Capability ID of 0015h is found for an extended capability, the test terminates with a failure.
6. For Base 2.x or later testing: if an Extended Capability ID of 0015h is found for an extended capability the following checks are performed on that extended capability structure:
7. The Capability Version field must be 1h.
8. The Next Capability Offset field must be 000h or greater than 0FFh and the lower 2 bits of this field must be 00b.
9. The following register field characteristic checks are performed:

Resizable BAR Extended Capability Header (Offset 00h) — DWORD

- | | |
|---------------------------------------|----|
| a. PCI Express Extended Capability ID | RO |
| b. Capability Version | RO |
| c. Next Capability Offset | RO |

10. A DWORD is read from offset 08h (Resizable BAR Control register 0).
11. The Number of Resizable BARs field (Resizable BAR Control register 0) must return a valid value (01h to 06h).
12. The number of resizable BARs [RBMax] is the value from the Number of Resizable BARs (Resizable BAR Control register 0) minus 1. The value [nBAR] will be used to denote the current resizable BAR number.
13. A DWORD is read from offset 04h (Resizable BAR Capability register 0).
14. Continue reading subsequent DWORDs until offset $([RBMax] * 08h) + 08h$ (Resizable BAR Control register [RBMax]) is read.
15. For each value of [nBAR] from 0 to [RBMax], the following register field characteristic checks are performed:

Resizable BAR Capability Register [nBAR] (Offset [nBAR] * 08h + 04h) — DWORD

- | | |
|----------------|---------|
| a. RsvdP_3-0 | RO-Zero |
| b. (bits 23-4) | RO |
| c. RsvdP_31-24 | RO-Zero |

Resizable BAR Control Register [nBAR] (Offset [nBAR] * 08h + 08h) — DWORD

- | | |
|--|---------|
| a. BAR Index | RO |
| b. RsvdP_4-3 | RO-Zero |
| c. Number of Resizable BARs
(for [nBAR] = 0) | RO |
| (for [nBAR] not 0) | RO-Zero |
| d. BAR Size
(only values corresponding to supported BAR sizes are used) | RW |
| e. RsvdP_31-13 | RO-Zero |

16. Set the current Resizable BAR number [nBAR] to 0.
17. Test software tests each Resizable BAR as follows:
 - a. Check that the Resizable BAR Capability Register [nBAR] returns a non-zero value. If not, the test fails and testing continues with the next Resizable BAR.
 - b. Check that the BAR Index field in Resizable BAR Control Register [nBAR] returns a valid value (0 to 5). If not, the test fails, and testing continues with the next Resizable BAR.
 - c. Check that the BAR location indicated by the value returned in the BAR Index field in Resizable BAR Control Register [nBAR] is a Memory Space BAR, by ensuring that the BAR's bit 0 returns 0.
 - d. Read the Type field, of the BAR location indicated by the value returned in the BAR Index field of the Resizable BAR Control Register [nBAR], and if it is 00b (32 bit BAR), then Resizable BAR Capability Register [nBAR] bits 23-16 must return 0. If not, then the test fails, and testing continues with the next Resizable BAR.
 - e. Test the resizable BAR controls as follows:
 - i. Set the value of Resizable BAR capability bit number [RBbit] to 4.
 - ii. Set the value of BAR size value [RBSize] to 0 0000b.
 - iii. If the Resizable BAR Capability Register [nBAR] bit [RBbit] returns a 1, then write the [RBSize] value to the BAR Size field of Resizable BAR Control Register [nBAR]. Otherwise go to step ix).
 - iv. Read back the BAR Size field of Resizable BAR Control Register [nBAR] and check that it returns the value written. If not, then the test fails and testing continues with the next Resizable BAR.
 - v. If the Type field, of the BAR location indicated by the value returned in the BAR Index field in the Resizable BAR Control Register [nBAR], is 00b (32 bit BAR), then read back that 32 bit BAR location and verify that bits ([RBbit] + 15) to 4 return 0. If not, then the test fails and testing continues with the next Resizable BAR. (This tests that BAR bits that are changed from RW to RO are automatically cleared to 0 by the hardware.)
 - vi. If the Type field, of the BAR location indicated by the value returned in the BAR Index field in the Resizable BAR Control Register [nBAR], is 10b (64 bit BAR), then read back that 64 bit BAR location (BAR Index returns the address of the lower 32 bits of the BAR) and verify that bits ([RBbit] + 15) to 4 return 0. If not, then the test fails and testing continues with the next Resizable BAR. (This tests that BAR bits that are changed from RW to RO are automatically cleared to 0 by the hardware.)
 - vii. If the Type field, of the BAR location indicated by the value returned in the BAR Index field in the Resizable BAR Control Register [nBAR], is 00b (32 bit BAR), then write FFFF FFFFh to that 32 bit BAR location. Read back the same 32 bit BAR location and verify that bits 31 to ([RBbit] + 16) return 1, and that bits ([RBbit] + 15) to 4 return 0. If not, then the test fails and testing continues with the next Resizable BAR.
 - viii. If the Type field, of the BAR location indicated by the value returned in the BAR Index field in the Resizable BAR Control Register [nBAR], is 10b (64 bit BAR), then write FFFF FFFF FFFF FFFFh to that 64 bit BAR location (BAR Index returns the address of the lower 32 bits of the BAR). Read back the same 64 bit BAR location and verify that bits 63 to ([RBbit] + 16) return 1, and that bits ([RBbit] + 15) to 4 return 0. If not, then the test fails and testing continues with the next Resizable BAR.
 - ix. Increment [RBbit] by 1. Increment [RBSize] by 1.
 - x. If [RBbit] is less than 24, then repeat steps iii) to x). If [RBbit] is 24, go to the next step.

18. Increment [nBAR] by 1. If the new value of [nBAR] is less than or equal to [RBMax], then repeat steps 17-18. Otherwise go to the next step.
19. For functions under test that have a link, the test is run at each of the following link speeds:
 - a. 2.5 GT/s (for devices that are capable). See Section 2.1.2.12 for details.
 - b. 5.0 GT/s (for devices that are capable). See Section 2.1.2.13 for details.
 - c. 8.0 GT/s (for devices that are capable). See Section 2.1.2.14 for details.
20. If the device type is Root Port or Root Complex Integrated Endpoint, then the test is repeated using each RCRB associated with the function under test, or associated with an RCRB that is itself associated with the function under test. See Section 2.1.2.17 for details. All offsets in the tests are now relative to the RCRB's base address and all cycles are now memory space accesses.

The test *fails* if:

- ☐ More than one Resizable BAR capability structure is present.
- ☐ An RCRB contains a Resizable BAR capability structure.
- ☐ The Capability Version field does not report 1h.
- ☐ The Next Capability Offset field does not report 000h or a value greater than 0FFh or the lower two bits are not 00b.
- ☐ The Number of Resizable BARs field in the Resizable BAR Control register 0 returns a reserved value.
- ☐ Any of the register field characteristic tests fail.
- ☐ Any implemented Resizable BAR Capability Register returns a zero value.
- ☐ Any implemented Resizable BAR Control Register returns a reserved value in the BAR Index field.
- ☐ Any implemented Resizable BAR Control Register returns a non-memory BAR value in the BAR Index field.
- ☐ Any implemented Resizable BAR Control Register returns a 32 bit memory BAR value in the BAR Index field, but the corresponding Resizable BAR Capability Register returns a value indicating support for 4 GB or greater BAR size.
- ☐ Any implemented Resizable BAR Control Register does not allow valid values to be written to the BAR Size field.
- ☐ Any implemented Resizable BAR Control Register does not force the corresponding BAR bits to be cleared to 0, when writing a value to the BAR Size field that will make the BAR bits change from RW to RO.
- ☐ Any implemented Resizable BAR Control Register does not force the corresponding BAR bits to become RO, when writing a value to the BAR Size field that will make the BAR bits change from RW to RO.

2.2.46. TD_1_46 Multicast Extended Capability Structure

The test verifies that if the function under test reports a Multicast (MC) Extended Capability structure, it is implemented as defined in the relevant specifications.

Relevant Specifications

- ☐ *PCI Express Base Specification*
- ☐ *ECN Multicast (to Base 2.0)*

(See Section 1.3 for additional specification revision details.)

Applicable Device/Port Types

This test is run on all device/port types.

This test is run on any RCRB associated with the function under test.

Starting Configuration

This test is run with functions starting in the D0-Uninitialized state following the standard initialization procedures in Section 2.1.1. Testing in additional Dx states may be optionally supported. ASPM states of the upstream and downstream ports are disabled for this test.

Overview of Test Steps

Test software performs the following steps:

1. Configure the function under test following the procedure described in Section 2.1.1.
2. Test software writes 0 to both the Memory Space Enable field and the Bus Master Enable field in the Command register (this disables responses to memory cycles targeting the function's BAR spaces and prevents the function from generating any requests).
3. Examine the Extended Capability ID field for each of the detected Extended Capabilities. Determine how many instances of the Extended Capability ID of 0012h (Multicast Extended Capability) are found. If more than one is found, the test terminates with a failure.
4. For Base 1.x testing: if the Extended Capability ID of 0012h is found for an extended capability, the test terminates with a failure.
5. For Base 2.x or later testing: if an Extended Capability ID of 0012h is found for an extended capability the following checks are performed on that extended capability structure:
6. The Capability Version field must be 1h.
7. The Next Capability Offset field must be 000h or greater than 0FFh and the lower 2 bits of this field must be 00b.
8. The device type must NOT be PCI Express to PCI/PCI-X Bridge or PCI/PCI-X to PCI Express Bridge.
9. A WORD is read from offset 04h (Multicast Capability register).
10. The number of MC Groups [MCGMax] is the value from the MC_Max_Group.
11. For Root Ports and Switches only: if the MC_ECRC_Regeneration_Supported field returns 1, then the function must support an AER Extended Capability and the function must have the ECRC Check Capable field (Advanced Error Capabilities and Control register) return 1.

12. The following register field characteristic checks are performed:

Multicast Extended Capability Header (Offset 00h) — DWORD

- | | |
|---------------------------------------|----|
| a. PCI Express Extended Capability ID | RO |
| b. Capability Version | RO |
| c. Next Capability Offset | RO |

Multicast Capability Register (Offset 04h) — WORD

- | | |
|--|---------|
| a. MC_Max_Group | |
| (For VFs) | RO-Zero |
| (otherwise) | RO |
| b. RsvdP_7-6 | RO-Zero |
| c. MC_Window_Size_Requested | |
| (For VFs) | RO-Zero |
| (for non-Root Ports, and non-Switches (not VFs)) | RO |
| (for Root Ports and Switches) | RO-Zero |
| d. RsvdP_14 | RO-Zero |
| e. MC_ECRC_Regeneration_Supported | |
| (For VFs, non-Root Ports, and non-Switches) | RO-Zero |
| (for Root Ports and Switches) | RO |

Multicast Control Register (Offset 06h) — WORD

- | | |
|-----------------|---------|
| a. MC_Num_Group | |
| (For VFs) | RO-Zero |
| (otherwise) | RW |
| b. RsvdP_14-6 | RO-Zero |
| c. MC_Enable | RW |

Multicast Base Address Register (Offset 08h) — 2 DWORDS

- | | |
|----------------------|---------|
| a. MC_Index_Position | |
| (For VFs) | RO-Zero |
| (otherwise) | RW |
| b. RsvdP_11-6 | RO-Zero |
| c. MC_Base_Address | |
| (For VFs) | RO-Zero |
| (otherwise) | RW |

MC_Receive Register (Offset 10h) — 2 DWORDS

- | | |
|---|---------|
| a. (bits [MCGMax] to 0) | RW |
| b. (if [MCGMax] is less than 63: bits 63 to [MCGMax] + 1) | RO-Zero |

MC_Block_All Register (Offset 18h) — 2 DWORDS

- | | |
|---|---------|
| a. (bits [MCGMax] to 0) | RW |
| b. (if [MCGMax] is less than 63: bits 63 to [MCGMax] + 1) | RO-Zero |

MC_Block_Untranslated Register (Offset 20h) — 2 DWORDS

- | | |
|---|---------------|
| a. (bits [MCGMax] to 0) | |
| (Root Ports and Switches) | RW |
| (Endpoint with ATS Extended Capability) | RW |
| (Endpoint without ATS Extended Capability) | RW or RO-Zero |
| b. (if [MCGMax] is less than 63: bits 63 to [MCGMax] + 1) | RO-Zero |

MC_Overlay_BAR Register (Offset 28h) — 2 DWORDS

(Root Port or Switch Port)

- | | |
|--------------------|----|
| a. MC_Overlay_Size | RW |
| b. MC_Overlay_BAR | RW |

13. The following default value checks are performed:

Multicast Control Register Default Value (Offset 06h) — WORD

- | | |
|-----------------|----------|
| a. MC_Num_Group | 00 0000b |
| b. MC_Enable | 0 |

Multicast Base Address Register Default Value (Offset 08h) — 2 DWORDS

- | | |
|----------------------|----------|
| a. MC_Index_Position | 00 0000b |
| b. MC_Base_Address | 0 |

MC_Receive Register Default Value (Offset 10h) — 2 DWORDS

- | | |
|----------------------------------|---|
| a. (Bits 0 through MC_Max_Group) | 0 |
|----------------------------------|---|

MC_Block_All Register Default Value (Offset 18h) — 2 DWORDS

- | | |
|----------------------------------|---|
| a. (Bits 0 through MC_Max_Group) | 0 |
|----------------------------------|---|

MC_Block_Untranslated Register Default Value (Offset 20h) — 2 DWORDS

- | | |
|----------------------------------|---|
| a. (Bits 0 through MC_Max_Group) | 0 |
|----------------------------------|---|

MC_Overlay_BAR Register Default Value (Offset 28h) — 2 DWORDS

(Root Port or Switch Port)

- | | |
|--------------------|----------|
| a. MC_Overlay_Size | 00 0000b |
| b. MC_Overlay_BAR | 0 |

14. For functions under test that have a link, the test is run at each of the following link speeds:

- 2.5 GT/s (for devices that are capable). See Section 2.1.2.12 for details.
- 5.0 GT/s (for devices that are capable). See Section 2.1.2.13 for details.
- 8.0 GT/s (for devices that are capable). See Section 2.1.2.14 for details.

15. If the device type is Root Port or Root Complex Integrated Endpoint, then the test is repeated using each RCRB associated with the function under test, or associated with an RCRB that is itself associated with the function under test. See Section 2.1.2.17 for details. All offsets in the tests are now relative to the RCRB's base address and all cycles are now memory space accesses.

The test *fails* if:

- ☐ More than one Multicast capability structure is present.
- ☐ The Capability Version field does not report 1h.
- ☐ The Next Capability Offset field does not report 000h or a value greater than 0FFh or the lower two bits are not 00b.
- ☐ The Multicast capability structure is present in a PCI Express to PCI/PCI-X Bridge or a PCI/PCI-X to PCI Express Bridge.
- ☐ The MC_ECRC_Regeneration_Supported field is 1, but the ECRC Check Capable field (AER Extended Capability) returns 0, or no AER Extended Capability is present.
- ☐ Any of the register field characteristic tests fail.
- ☐ Any of the default value tests fail.

2.2.47. TD_1_47 LTR Extended Capability Structure

The test verifies that if the function under test reports a Latency Tolerance Reporting (LTR) Extended Capability structure, it is implemented as defined in the relevant specifications.

Relevant Specifications

- ❑ *PCI Express Base Specification*
- ❑ *ECN Latency Tolerance Reporting (to Base 2.0)*

(See Section 1.3 for additional specification revision details.)

Applicable Device/Port Types

This test is run on all device/port types.

This test is run on any RCRB associated with the function under test.

Starting Configuration

This test is run with functions starting in the D0-Uninitialized state following the standard initialization procedures in Section 2.1.1. Testing in additional Dx states may be optionally supported. ASPM states of the upstream and downstream ports are disabled for this test.

Overview of Test Steps

Test software performs the following steps:

1. Configure the function under test following the procedure described in Section 2.1.1.
2. Examine the Extended Capability ID field for each of the detected Extended Capabilities. Determine how many instances of the Extended Capability ID of 0018h (LTR Extended Capability) are found. If more than one is found, the test terminates with a failure.
3. If the Extended Capability ID of 0018h is found in an RCRB, the test terminates with a failure.
4. For Base 1.x testing: if the Extended Capability ID of 0018h is found for an extended capability, the test terminates with a failure.
5. For Base 2.x or later testing: if an Extended Capability ID of 0018h is found for an extended capability the following checks are performed on that extended capability structure:
6. The Capability Version field must be 1h.
7. The Next Capability Offset field must be 000h or greater than 0FFh and the lower 2 bits of this field must be 00b.
8. The device type must be an Endpoint or a Switch Upstream Port.
9. The function number must be 0.
10. The following register field characteristic checks are performed:

LTR Extended Capability Header (Offset 00h) — DWORD

- | | |
|---------------------------------------|----|
| a. PCI Express Extended Capability ID | RO |
| b. Capability Version | RO |
| c. Next Capability Offset | RO |

Max Snoop Latency Register (Offset 04h) — WORD

- | | |
|--|---------|
| a. Max Snoop Latency Value | RW |
| b. Max Snoop Latency Scale | RW |
| (test software only writes permitted values of 000b to 101b) | |
| c. RsvdP_15-13 | RO-Zero |

Max No-Snoop Latency Register (Offset 06h) — WORD

- | | |
|--|---------|
| a. Max No-Snoop LatencyValue | RW |
| b. Max No-Snoop LatencyScale | RW |
| (test software only writes permitted values of 000b to 101b) | |
| c. RsvdP_15-13 | RO-Zero |

11. The following default value checks are performed:

Max Snoop Latency Register Default Value (Offset 04h) — WORD

- | | |
|----------------------------|------|
| a. Max Snoop Latency Value | 0 |
| b. Max Snoop Latency Scale | 000b |

Max No-Snoop Latency Register Default Value (Offset 06h) — WORD

- | | |
|------------------------------|------|
| a. Max No-Snoop LatencyValue | 0 |
| b. Max No-Snoop LatencyScale | 000b |

12. For functions under test that have a link, the test is run at each of the following link speeds:

- a. 2.5 GT/s (for devices that are capable). See Section 2.1.2.12 for details.
- b. 5.0 GT/s (for devices that are capable). See Section 2.1.2.13 for details.
- c. 8.0 GT/s (for devices that are capable). See Section 2.1.2.14 for details.

13. If the device type is Root Port or Root Complex Integrated Endpoint, then the test is repeated using each RCRB associated with the function under test, or associated with an RCRB that is itself associated with the function under test. See Section 2.1.2.17 for details. All offsets in the tests are now relative to the RCRB's base address and all cycles are now memory space accesses.

The test *fails* if:

- ☐ More than one LTR capability structure is present.
- ☐ An RCRB contains a LTR capability structure.
- ☐ The Capability Version field does not report 1h.
- ☐ The Next Capability Offset field does not report 000h or a value greater than 0FFh or the lower two bits are not 00b.
- ☐ The LTR capability structure is present in any function other than the following: Endpoint; Legacy Endpoint; Root Complex Integrated Endpoint; Switch Upstream Port.
- ☐ The LTR capability is present in a function other than function 0.
- ☐ Any of the register field characteristic tests fail.
- ☐ Any of the default value tests fail.

2.2.48. TD_1_48 TPH Requester Extended Capability Structure

The test verifies that if the function under test reports a TLP Processing Hints (TPH) Requester Extended Capability structure, it is implemented as defined in the relevant specifications.

Relevant Specifications

- ❑ *PCI Express Base Specification*
- ❑ *ECN TLP Processing Hints (to Base 2.0)*

(See Section 1.3 for additional specification revision details.)

Applicable Device/Port Types

This test is run on all device/port types.

This test is run on any RCRB associated with the function under test.

Starting Configuration

This test is run with functions starting in the D0-Uninitialized state following the standard initialization procedures in Section 2.1.1. Testing in additional Dx states may be optionally supported. ASPM states of the upstream and downstream ports are disabled for this test.

Overview of Test Steps

Test software performs the following steps:

1. Configure the function under test following the procedure described in Section 2.1.1.
2. Test software writes 0 to the Bus Master Enable field in the Command register (this prevents the function from generating any requests).
3. Examine the Extended Capability ID field for each of the detected Extended Capabilities. Determine how many instances of the Extended Capability ID of 0017h (TPH Requester Extended Capability) are found. If more than one is found, the test terminates with a failure.
4. If the Extended Capability ID of 0017h is found in an RCRB, the test terminates with a failure.
5. For Base 1.x testing: if the Extended Capability ID of 0017h is found for an extended capability, the test terminates with a failure.
6. For Base 2.x or later testing: if an Extended Capability ID of 0017h is found for an extended capability the following checks are performed on that extended capability structure:
 - a. The Capability Version field must be 1h.
 - b. The Next Capability Offset field must be 000h or greater than 0FFh and the lower 2 bits of this field must be 00b.
 - c. The device type must not be PCI Express to PCI/PCI-X Bridge or PCI/PCI-X to PCI Express Bridge.

7. Test software reads a DWORD from offset 04h (TPH Requester Capability register) and performs the following checks:
 - a. The No ST Mode Supported field must be 1.
 - b. If the Interrupt Vector Mode Supported field reports 1, then the ST Table Location field must be 01b or 10b.
 - c. If both the Interrupt Vector Mode Supported field and the Device Mode Supported field report 0, then the ST Table Location field must be 00b.
 - d. The ST Table Location field must report a valid value (00b, 01b, 10b).
 - e. If the ST Table Location field reports 01b, then the ST Table Size field must be less than 64 (000 0100 0000b).
 - f. If the ST Table Location field reports 10b, then the function must implement a MSI-X Capability.

8. The following register field characteristic checks are performed:

TPH Requester Extended Capability Header (Offset 00h) — DWORD

- | | |
|---------------------------------------|----|
| a. PCI Express Extended Capability ID | RO |
| b. Capability Version | RO |
| c. Next Capability Offset | RO |

TPH Requester Capability Register (Offset 04h) — DWORD

- | | |
|-------------------------------------|---------|
| a. No ST Mode Supported | RO |
| b. Interrupt Vector Mode Supported | RO |
| c. Device Specific Mode Supported | RO |
| d. RsvdP_7-3 | RO-Zero |
| e. Extended TPH Requester Supported | RO |
| f. ST Table Location | RO |
| g. RsvdP_15-11 | RO-Zero |
| h. ST Table Size | RO |
| i. RsvdP_31-27 | RO-Zero |

TPH Requester Control Register (Offset 08h) — DWORD

- | | |
|---|-------------------|
| a. ST Mode Select
(If both Interrupt Vector Mode Supported
and Device Specific Mode Supported are 0)
(If either Interrupt Vector Mode Supported
or Device Specific Mode supported are non-zero) | RO-Zero

RW |
| b. RsvdP_7-3 | RO-Zero |
| c. TPH Requester Enable | RW |
| d. RsvdP_31-10 | RO-Zero |

9. The following default value checks are performed:

TPH Requester Control Register Default Value (Offset 08h) — DWORD

- | | |
|-------------------------|------|
| a. ST Mode Select | 000b |
| b. TPH Requester Enable | 00b |

10. If ST Table Location is 01b, the following is performed:
- Calculate the ST Table Size value [STS] as follows: [STS] = the value returned in the ST Table Size field.
 - Every TPH ST Table Entry is a WORD, with the lower 16 bits of the read DWORD as the lower entry.
 - Test software reads DWORDs starting from offset 0Ch (TPH ST Table) until the DWORD that contains the last TPH ST Table Entry is read.
 - For each TPH ST Table Entry value [n], the following register field characteristic checks are performed:

ST Table Entries Registers (n) (Offset 0Ch + 2 * n) — WORD	
a. ST Lower	RW
b. ST Upper	
(If Extended TPH Requester Supported is 1)	RW
(If Extended TPH Requester Supported is 0)	RO-Zero
 - For each TPH ST Table Entry value [n], the following default value checks are performed:

ST Table Entries Registers (n) Default Value (Offset 0Ch + 2 * n) — WORD	
a. ST Lower	00h
b. ST Upper	00h
 - If [STS] is an even number, then the upper 16 bits of the last read DWORD (Offset 0Ch + 2 * [STS]) must be RO-Zero.
11. For functions under test that have a link, the test is run at each of the following link speeds:
- 2.5 GT/s (for devices that are capable). See Section 2.1.2.12 for details.
 - 5.0 GT/s (for devices that are capable). See Section 2.1.2.13 for details.
 - 8.0 GT/s (for devices that are capable). See Section 2.1.2.14 for details.
12. If the device type is Root Port or Root Complex Integrated Endpoint, then the test is repeated using each RCRB associated with the function under test, or associated with an RCRB that is itself associated with the function under test. See Section 2.1.2.17 for details. All offsets in the tests are now relative to the RCRB's base address and all cycles are now memory space accesses.

The test *fails* if:

- ☐ More than one TPH Requester capability structure is present.
- ☐ An RCRB contains a TPH Requester capability structure.
- ☐ The Capability Version field does not report 1h.
- ☐ The Next Capability Offset field does not report 000h or a value greater than 0FFh or the lower two bits are not 00b.
- ☐ The TPH capability structure is present in a Bridge.
- ☐ The function does not report the No ST Mode Supported field as 1.
- ☐ The function reports the Interrupt Vector Mode Supported field as 1, but does not report that ST Table Location is in the TPH Requester Capability or in the MSI-X Table.
- ☐ The function reports the ST Table Location as in either the MSI-X Table or the TPH Requester Capability, but does not report either the Interrupt Vector Mode Supported field as 1 or the Device Specific Mode Supported field as 1.

- ❑ The function reports an invalid ST Table Location field value.
- ❑ The function reports the ST Table Location as in the TPH Requester Capability, but reports an invalid ST Table Size field (greater than 64 ST Table Entries).
- ❑ The function reports the ST Table Location as in the MSI-X Table, but does not implement a MSI-X Capability.
- ❑ Any of the register field characteristic tests fail.
- ❑ Any of the default value tests fail.

2.2.49. TD_1_54 ATS Extended Capability Structure

The test verifies that if the function under test reports an Address Translation Services (ATS) Extended Capability structure, it is implemented as defined in the relevant specifications.

Relevant Specifications

- ❑ *Address Translation Services Revision 1.1*
- ❑ *Single Root I/O Virtualization and Sharing Specification*

(See Section 1.3 for additional specification revision details.)

Applicable Device/Port Types

This test is run on all device/port types.

This test is run on any RCRB associated with the function under test.

Starting Configuration

This test is run with functions starting in the D0-Uninitialized state following the standard initialization procedures in Section 2.1.1. Testing in additional Dx states may be optionally supported. ASPM states of the upstream and downstream ports are disabled for this test.

Overview of Test Steps

Test software performs the following steps:

1. Configure the function under test following the procedure described in Section 2.1.1.
2. Test software writes 0 to the Bus Master Enable field in the Command register (this prevents the function from generating any requests).
3. Examine the Extended Capability ID field for each of the detected Extended Capabilities. Determine how many instances of the Extended Capability ID of 000Fh (ATS Extended Capability) are found. If more than one is found, the test terminates with a failure.
4. If the Extended Capability ID of 000Fh is found in an RCRB, the test terminates with a failure.
5. If an Extended Capability ID of 000Fh is found for an extended capability, the following checks are performed on that extended capability structure:
 - a. The Capability Version field must be 1h.
 - b. The Next Capability Offset field must be 000h or greater than 0FFh and the lower 2 bits of this field must be 00b.
 - c. The device type must be PCI Express Endpoint, Legacy Endpoint, or Root Complex Integrated Endpoint.
6. The following register field characteristic checks are performed:

ATS Extended Capability Header (Offset 00h) — DWORD

- | | |
|---------------------------------------|----|
| a. PCI Express Extended Capability ID | RO |
| b. Capability Version | RO |
| c. Next Capability Offset | RO |

ATS Capability Register (Offset 04h) — WORD

- | | |
|--|---------|
| a. Invalidate Queue Depth
(for VFs) | RO-Zero |
| (otherwise) | RO |
| b. Page Aligned Request | RO |
| c. RsvdZ_15-6 | RO-Zero |

ATS Control Register (Offset 06h) — WORD

- | | |
|---|---------|
| a. Smallest Translation Unit (STU)
(for VFs) | RO-Zero |
| (otherwise) | RW |
| b. RsvdP_14-5 | RO-Zero |
| c. Enable | RW |

7. The following default value checks are performed:

ATS Control Register Default Value (Offset 06h) — WORD

- | | |
|------------------------------------|---------|
| a. Smallest Translation Unit (STU) | 0 0000b |
| b. Enable | 0 |

8. For functions under test that have a link, the test is run at each of the following link speeds:
 - a. 2.5 GT/s (for devices that are capable). See Section 2.1.2.12 for details.
 - b. 5.0 GT/s (for devices that are capable). See Section 2.1.2.13 for details.
 - c. 8.0 GT/s (for devices that are capable). See Section 2.1.2.14 for details.

9. If the device type is Root Port or Root Complex Integrated Endpoint, then the test is repeated using each RCRB associated with the function under test, or associated with an RCRB that is itself associated with the function under test. See Section 2.1.2.17 for details. All offsets in the tests are now relative to the RCRB's base address and all cycles are now memory space accesses.

The test *fails* if:

- ☐ More than one ATS capability structure is present.
- ☐ An RCRB contains an ATS capability structure.
- ☐ The Capability Version field does not report 1h.
- ☐ The Next Capability Offset field does not report 000h or a value greater than 0FFh or the lower two bits are not 00b.
- ☐ The ATS capability structure is present in a non-Endpoint.
- ☐ Any of the register field characteristic tests fail.
- ☐ Any of the default value tests fail.

2.2.50. TD_1_55 Page Request Extended Capability Structure

The test verifies that if the function under test reports a Page Request Extended Capability structure, it is implemented as defined in the relevant specifications.

Relevant Specifications

- ☐ *Address Translation Services Revision 1.1*

(See Section 1.3 for additional specification revision details.)

Applicable Device/Port Types

This test is run on all device/port types.

This test is run on any RCRB associated with the function under test.

Starting Configuration

This test is run with functions starting in the D0-Uninitialized state following the standard initialization procedures in Section 2.1.1. Testing in additional Dx states may be optionally supported. ASPM states of the upstream and downstream ports are disabled for this test.

Overview of Test Steps

Test software performs the following steps:

1. Configure the function under test following the procedure described in Section 2.1.1.
2. Test software writes 0 to the Bus Master Enable field in the Command register (this prevents the function from generating any requests).
3. Examine the Extended Capability ID field for each of the detected Extended Capabilities. Determine how many instances of the Extended Capability ID of 0013h (Page Request Extended Capability) are found. If more than one is found, the test terminates with a failure.
4. If the Extended Capability ID of 0013h is found in an RCRB, the test terminates with a failure.

5. If an Extended Capability ID of 0013h is found for an extended capability, the following checks are performed on that extended capability structure:
 - a. The Capability Version field must be 1h.
 - b. The Next Capability Offset field must be 000h or greater than 0FFh and the lower 2 bits of this field must be 00b.
 - c. The device type must be PCI Express Endpoint, Legacy Endpoint, or Root Complex Integrated Endpoint.
6. The following register field characteristic checks are performed:

Page Request Extended Capability Header (Offset 00h) — DWORD

- | | |
|---------------------------------------|----|
| a. PCI Express Extended Capability ID | RO |
| b. Capability Version | RO |
| c. Next Capability Offset | RO |

Page Request Control Register (Offset 04h) — WORD

- | | |
|---------------|---------|
| a. Enable | RW |
| b. Reset | RO-Zero |
| c. RsvdP_15-2 | RO-Zero |

Page Request Status Register (Offset 06h) — WORD

- | | |
|---|---------|
| a. Response Failure | RW1C |
| b. Unexpected Page Request Group Index | RW1C |
| c. RsvdZ_7-2 | RO-Zero |
| d. Stopped | RO |
| (The Enable field must be written with 0 before this field is tested) | |
| e. RsvdZ_15-9 | RO-Zero |

Outstanding Page Request Capacity Register (Offset 08h) — DWORD

(all bits)	RO
------------	----

Outstanding Page Request Allocation Register (Offset 0Ch) — DWORD

(all bits)	RW
(The Enable field must be written with 0 before this register is tested. Only values up to the value returned in the Outstanding Page Request Capacity register can be written.)	

7. The following default value checks are performed:

Page Request Control Register Default Value (Offset 04h) — WORD

- | | |
|-----------|---|
| a. Enable | 0 |
|-----------|---|

Page Request Status Register Default Value (Offset 06h) — WORD

- | | |
|--|---|
| a. Response Failure | 0 |
| b. Unexpected Page Request Group Index | 0 |
| c. Stopped | 1 |

Outstanding Page Request Allocation Register Default Value (Offset 0Ch) — DWORD

(This register defaults to 0000 0000h.)

8. For functions under test that have a link, the test is run at each of the following link speeds:
 - a. 2.5 GT/s (for devices that are capable). See Section 2.1.2.12 for details.
 - b. 5.0 GT/s (for devices that are capable). See Section 2.1.2.13 for details.
 - c. 8.0 GT/s (for devices that are capable). See Section 2.1.2.14 for details.

9. If the device type is Root Port or Root Complex Integrated Endpoint, then the test is repeated using each RCRB associated with the function under test, or associated with an RCRB that is itself associated with the function under test. See Section 2.1.2.17 for details. All offsets in the tests are now relative to the RCRB's base address and all cycles are now memory space accesses.

The test *fails* if:

- ☐ More than one Page Request capability structure is present.
- ☐ An RCRB contains a Page Request capability structure.
- ☐ The Capability Version field does not report 1h.
- ☐ The Next Capability Offset field does not report 000h or a value greater than 0FFh or the lower two bits are not 00b.
- ☐ The Page Request capability structure is present in a non-Endpoint.
- ☐ Any of the register field characteristic tests fail.
- ☐ Any of the default value tests fail.

2.2.51. TD_1_56 SR-IOV Extended Capability Structure

The test verifies that if the function under test reports a SR-IOV Extended Capability structure, it is implemented as defined in the relevant specifications.

Relevant Specifications

- ☐ *Single Root I/O Virtualization and Sharing Specification*

(See Section 1.3 for additional specification revision details.)

Applicable Device/Port Types

This test is run on all device/port types.

This test is run on any RCRB associated with the function under test.

Starting Configuration

This test is run with functions starting in the D0-Uninitialized state following the standard initialization procedures in Section 2.1.1. Testing in additional Dx states may be optionally supported. ASPM states of the upstream and downstream ports are disabled for this test.

Overview of Test Steps

Test software performs the following steps:

1. Configure the function under test following the procedure described in Section 2.1.1.
2. Examine the Extended Capability ID field for each of the detected Extended Capabilities. Determine how many instances of the Extended Capability ID of 0010h (SR-IOV Extended Capability) are found. If more than one is found, the test terminates with a failure.
3. If the Extended Capability ID of 0010h is found in an RCRB, the test terminates with a failure.

4. If an Extended Capability ID of 0010h is found for an extended capability, the following checks are performed on that extended capability structure:
 - a. The Capability Version field must be 1h.
 - b. The Next Capability Offset field must be 000h or greater than 0FFh and the lower 2 bits of this field must be 00b.
 - c. The device type must be PCI Express Endpoint, Legacy Endpoint, or Root Complex Integrated Endpoint.
5. Test software reads a WORD from offset 0Eh (TotalVFs register).
6. Test software reads a WORD from offset 0Ch (InitialVFs register) and performs the following checks:
 - a. The InitialVFs register value must be less than or equal to the TotalVFs register value.
7. For Base 3.x or later testing: test software reads a WORD from offset 10h (NumVFs register) and if it returns a non-zero value, the following checks are performed:
 - a. Test software reads a WORD from offset 14h (First VF Offset register) and if a value of 0000h is returned the test fails.
8. Test software reads a DWORD from offset 1Ch (Supported Page Sizes register) and performs the following checks:
 - a. The Supported Page Sizes register value must have at least all the following bits return 1: (bit 10; bit 8; bit 6; bit 4; bit 1; bit 0).
9. For the DWORD from offset 20h (System Page Size register) test software performs the following checks:
 - a. Test software writes the VF Enable field to 0.
 - b. Set value [NBIT] to 0.
 - c. If bit [NBIT] in the Supported Page Sizes register returns 1, then write a DWORD to the System Page Size register with bit [NBIT] set to 1, and all bits other than [NBIT] set to 0.
 - d. If bit [NBIT] in the Supported Page Sizes register returns 1, read back the DWORD from the System Page Size and verify that bit [NBIT] returns 1 and all bits other than [NBIT] return 0. If not, the test fails.
 - e. Increment [NBIT] by 1 and repeat steps c-e) for all values of [NBIT] up to and including 31.
10. Starting at offset 24h the next 6 DWORDs are read. These are the VF BAR0-VF BAR5 registers. (Some VF BAR registers may occupy 2 DWORDs if they are 64 bit VF BARs.)
 - a. Each VF BAR n must have bit 0 = 0 (Memory Space BAR).
 - b. Offset 38h (VF BAR 5) must not have both bit 0 = 0 (Memory Space BAR) and the Type field = 10b (64 bit addressing).
11. Test software writes the VF MSE field to 0 (to disable all VF memory space).
12. For each VF BAR test software writes FFFF FFFFh to the VF BAR and then reads back the same register. If it returns 0000 0000h, then this is an empty VF BAR. Empty VF BARs are excluded from further testing.
13. For each non-empty VF BAR:
 - a. Test software writes the VF Enable field to 0.
 - b. Set value [NBIT] to 0.
 - c. If bit [NBIT] in the Supported Page Sizes register returns 0, then skip to step m)
 - d. If bit [NBIT] in the Supported Page Sizes register returns 1, then write a DWORD to the System Page Size register with bit [NBIT] set to 1, and all bits other than [NBIT] set to 0.

- e. Test software writes FFFF FFFFh to the VF BAR and then reads back the same register. Bits 6-4 must be 0 (MEMORY BAR size must be 128 Bytes or larger).
- f. If the Type field returns 00b, then starting at bit 4 determine the least significant bit value [LSB] that is set to 1. [LSB] must be less than or equal to 31. All bits from 31 to [LSB] must also be 1. This same [LSB] value is used in subsequent tests of this same VF BAR.
- g. If the Type field returns 10b, test software writes FFFF FFFFh to the immediate next higher Configuration Space location (this is the upper half of a 64 bit VF BAR) then starting at bit 4 of the lower 32 bit VF BAR determine the least significant bit value [LSB] that is set to 1. [LSB] must be less than or equal to 63. All bits from 63 to [LSB] must also be 1.
- h. [LSB] must be equal or greater than $([NBIT] + 12)$ (VF BAR must be aligned to the current System Page Size boundary setting.)
- i. Test software writes 5555 5555h (for 32 bit VF BAR) or 5555 5555 5555 5555h (for 64 bit VF BAR) to the VF BAR and then reads back the same register. Bits 31-[LSB] (for 32 bit VF BAR) or Bits 63-[LSB] (for 64 bit VF BAR) must return the written value. Bits [LSB]-4 must return all 0.
- j. Test software writes AAAA AAAAh (for 32 bit VF BAR) or AAAA AAAA AAAA AAAAh (for 64 bit VF BAR) to the VF BAR and then reads back the same register. Bits 31-[LSB] (for 32 bit VF BAR) or Bits 63-[LSB] (for 64 bit VF BAR) must return the written value. Bits [LSB]-4 must return all 0.

- k. Test software writes CCCC CCCC_h (for 32 bit VF BAR) or CCCC CCCC CCCC CCCC_h (for 64 bit VF BAR) to the VF BAR and then reads back the same register. Bits 31-[LSB] (for 32 bit VF BAR) or Bits 63-[LSB] (for 64 bit VF BAR) must return the written value. Bits [LSB]-4 must return all 0.
 - l. Test software writes 0000 0000_h (for 32 bit VF BAR) or 0000 0000 0000 0000_h (for 64 bit VF BAR) to the VF BAR and then reads back the same register. Bits 31-[LSB] (for 32 bit VF BAR) or Bits 63-[LSB] (for 64 bit VF BAR) must return the written value. Bits [LSB]-4 must return all 0.
 - m. Increment [NBIT] by 1 and repeat steps c-m) for all values of [NBIT] up to and including 31.
14. If the TotalVFs register returns a value greater than 0 and the VF Migration Capable field (SR-IOV Capabilities register) returns 1, test software reads a DWORD from offset 3Ch (VF Migration State Array Offset register) and performs the following checks:
- a. The VF Migration State BIR field must return one of the following values: 000b to 101b.
 - b. The BAR pointed to by VF Migration State BIR field value is read and bit 0 of that BAR must be 0 (Memory BAR).
15. The following register field characteristic checks are performed:

SR-IOV Extended Capability Header (Offset 00h) — DWORD

- | | |
|---------------------------------------|----|
| a. PCI Express Extended Capability ID | RO |
| b. Capability Version | RO |
| c. Next Capability Offset | RO |

SR-IOV Capabilities Register (Offset 04h) — DWORD

- | | |
|--|---------------|
| a. VF Migration Capable | RO |
| b. ARI Capable Hierarchy Preserved
(For Base 2.x or later testing)
(for lowest numbered PF in the device)
(for all other PFs in the device) | RO
RO-Zero |
| c. RsvdP_20-2
(For Base 2.x or later testing) | RO-Zero |
| d. RsvdP_20-1
(For Base 1.x testing) | RO-Zero |
| e. VF Migration Interrupt Message Number | RO |

SR-IOV Control Register (Offset 08h) — WORD

- | | |
|---|----------------------|
| a. VF Enable | RW |
| b. VF Migration Enable
(if VF Migration Capable is 1 and VF Enable is written with 0)
(if VF Migration Capable is 1 and VF Enable is written with 1)
(if VF Migration Capable is 0)
(The VF Enable field must be written with 0 before this field is tested as RW. When the VF Enable field is written with 1, the field will become RO.) | RW
RO
RO-Zero |
| c. VF Migration Interrupt Enable
(For Base 1.x testing)
(For Base 2.x or later testing: if VF Migration Capable is 1)
(For Base 2.x or later testing: if VF Migration Capable is 0) | RW
RW
RW or RO |

- | | |
|--|--------------------------|
| d. VF MSE | RW |
| e. ARI Capable Hierarchy
(for RC Integrated Endpoints)
(for lowest numbered PF in the device)
(for all other PFs in the device) | RO-Zero
RW
RO-Zero |
| f. RsvdP_15-5 | RO-Zero |

SR-IOV Status Register (Offset 0Ah) — WORD

- | | |
|------------------------|---------|
| a. VF Migration Status | RW1C |
| b. RsvdZ_15-1 | RO-Zero |

InitialVFs Register (Offset 0Ch) — WORD

RO

TotalVFs Register (Offset 0Eh) — WORD

RO

NumVFs Register (Offset 10h) — WORD

RW

(The VF Enable field must be written with 0 before this register is tested. Only values up to the value returned in the TotalVFs register can be written.)

Function Dependency Link Register (Offset 12h) — BYTE

RO

Reserved Register (Offset 13h) — BYTE

- | | |
|--------------|---------|
| a. RsvdP_7-0 | RO-Zero |
|--------------|---------|

First VF Offset Register (Offset 14h) — WORD

RO

VF Stride Register (Offset 16h) — WORD

RO

Reserved Register (Offset 18h) — WORD

- | | |
|---------------|---------|
| a. RsvdP_15-0 | RO-Zero |
|---------------|---------|

VF Device ID Register (Offset 1Ah) — WORD

RO

Supported Page Sizes Register (Offset 1Ch) — DWORD

(all bits)

RO

VF BAR (n) (lower 32) Register (Offset 24h + n * 04h) — DWORD

(for non-empty VF BARs)

- | | |
|-----------------|---------|
| a. bit 0 | RO-Zero |
| b. Type | RO |
| c. Prefetchable | RO |

Values of [n] between 0 and 5 are to be tested. If Type field of VF BAR (n) is 10b (64 bit addressing), then increment [n] by 2, otherwise increment [n] by 1.

VF Migration State Array Offset Register (Offset 3Ch) — DWORD

- | | |
|---|---------------|
| a. VF Migration State BIR
(if TotalVFs is greater than 0 and VF Migration Capable is 1)
(if VF Migration Capable is 0) | RO
RO-Zero |
| b. VF Migration State Offset
(if TotalVFs is greater than 0 and VF Migration Capable is 1)
(if VF Migration Capable is 0) | RO
RO-Zero |

VF Migration State Array ([table] + [n-1]) — BYTE

(only if TotalVFs is greater than 0 and VF Migration Capable is 1)

- | | |
|-----------------------|---------|
| a. VF Migration State | RW |
| b. RsvdP_7-2 | RO-Zero |

(The NumVFs register must be written with the value returned by the TotalVFs register, and then the VF Enable field must be written with 1 before this register is tested.)

Note: Testing this requires programming the designated Memory BAR and enabling Memory space. The value of [table] is the BAR contents (pointed to by VF Migration State BIR field) plus the offset (value in the VF Migration State Offset field multiplied by 8). All values of [n-1] between 0 and the value in the NumVFs field minus 1 must be tested.

16. The following default value checks are performed:

SR-IOV Control Register Default Value (Offset 08h) — WORD

- | | |
|----------------------------------|---|
| a. VF Enable | 0 |
| b. VF Migration Enable | 0 |
| c. VF Migration Interrupt Enable | 0 |
| d. VF MSE | 0 |
| e. ARI Capable Hierarchy | 0 |

SR-IOV Status Register Default Value (Offset 0Ah) — WORD

- | | |
|------------------------|---|
| a. VF Migration Status | 0 |
|------------------------|---|

System Page Size Register Default Value (Offset 20h) — DWORD

(This register defaults to 0000 0001h.)

VF Migration State Array Default Value ([table] + [n-1]) — BYTE

(only if TotalVFs is greater than 0 and VF Migration Capable is 1)

- | | |
|--|-----|
| a. VF Migration State | |
| (for [n-1] less than or equal to InitialVFs value minus 1) | 11b |
| (for [n-1] greater than InitialVFs value minus 1) | 00b |

(After returning the function to its default state test software must wait 1 second, then it must write the NumVFs register with the value returned by the TotalVFs register, and then it must write the VF Enable field with 1 before this array is read.)

Note: Testing this requires programming the designated Memory BAR and enabling Memory space. The value of [table] is the BAR contents (pointed to by the VF Migration State BIR field value) plus the offset (value in the VF Migration State Offset field multiplied by 8). All values of [n-1] between 0 and the value in the NumVFs field minus 1 must be tested.

17. For functions under test that have a link, the test is run at each of the following link speeds:
- 2.5 GT/s (for devices that are capable). See Section 2.1.2.12 for details.
 - 5.0 GT/s (for devices that are capable). See Section 2.1.2.13 for details.
 - 8.0 GT/s (for devices that are capable). See Section 2.1.2.14 for details.
18. If the device type is Root Port or Root Complex Integrated Endpoint, then the test is repeated using each RCRB associated with the function under test, or associated with an RCRB that is itself associated with the function under test. See Section 2.1.2.17 for details. All offsets in the tests are now relative to the RCRB's base address and all cycles are now memory space accesses.

The test *fails* if:

- ☐ More than one SR-IOV capability structure is present.
- ☐ An RCRB contains a SR-IOV capability structure.
- ☐ The Capability Version field does not report 1h.
- ☐ The Next Capability Offset field does not report 000h or a value greater than 0FFh or the lower two bits are not 00b.
- ☐ The SR-IOV capability structure is present in a non-Endpoint.
- ☐ The InitialVFs register is greater than the TotalVFs register.
- ☐ The First VF Offset register is 0000h when the NumVFs register is greater than zero (for Base 3.x or later testing only).
- ☐ The Supported Page Sizes register does not have one of the following bits set: (bit 10; bit 8; bit 6; bit 4; bit 1, bit 0).
- ☐ The System Page Size register does not have all bits writeable that correspond to bits that return 1 in the Supported Page Sizes register.
- ☐ A VF BAR claims less than 128 Bytes of memory space.
- ☐ A VF BAR is not aligned to the current System Page Size boundary setting.
- ☐ A VF BAR does not return contiguous 1's across all implemented address bits.
- ☐ A VF BAR does not implement RW bits across all implemented address bits.
- ☐ The VF Migration State BIR field returns a reserved value, when the TotalVFs register is greater than 0 and the VF Migration Capable field is 1.
- ☐ The VF Migration State BIR field value points to an I/O Space BAR, when the TotalVFs register is greater than 0 and the VF Migration Capable field is 1.
- ☐ Any of the register field characteristic tests fail.
- ☐ Any of the default value tests fail.

2.2.52. TD_1_57 MR-IOV Extended Capability Structure

The test verifies that if the function under test reports a MR-IOV Extended Capability structure, it is implemented as defined in the relevant specifications.

Only a brief sub-set of MR-IOV testing will be implemented in the compliance test suite at this time.

Note: A comprehensive test description is provided in Section A.1 in Appendix A, for informational purposes only.

Relevant Specifications

- ☐ *Multi-Root I/O Virtualization and Sharing Specification Revision 1.0*

(See Section 1.3 for additional specification revision details.)

Applicable Device/Port Types

This test is run on all device/port types.

This test is run on any RCRB associated with the function under test.

Starting Configuration

This test is run with functions starting in the D0-Uninitialized state following the standard initialization procedures in Section 2.1.1. Testing in additional Dx states may be optionally supported. ASPM states of the upstream and downstream ports are disabled for this test.

Overview of Test Steps

Test software performs the following steps:

1. Configure the function under test following the procedure described in Section 2.1.1.
2. Examine the Extended Capability ID field for each of the detected Extended Capabilities. Determine how many instances of the Extended Capability ID of 0011h (MR-IOV Extended Capability) are found. If more than one is found, the test terminates with a failure.
3. If the Extended Capability ID of 0011h is found in an RCRB, the test terminates with a failure.
4. If an Extended Capability ID of 0011h is found for an extended capability, the following checks are performed on that extended capability structure:
 - a. The Capability Version field must be 1h.
 - b. The Next Capability Offset field must be 000h or greater than 0FFh and the lower 2 bits of this field must be 00b.
 - c. The device type must be PCI Express Endpoint, Legacy Endpoint, Root Complex Integrated Endpoint, Switch Upstream Port, or Switch Downstream Port.
 - d. If the device type is PCI Express Endpoint, Legacy Endpoint, or Root Complex Integrated Endpoint the following checks are performed:
 - i. Must not be a VF.
5. For functions under test that have a link, the test is run at each of the following link speeds:
 - a. 2.5 GT/s (for devices that are capable). See Section 2.1.2.12 for details.
 - b. 5.0 GT/s (for devices that are capable). See Section 2.1.2.13 for details.
 - c. 8.0 GT/s (for devices that are capable). See Section 2.1.2.14 for details.
6. If the device type is Root Port or Root Complex Integrated Endpoint, then the test is repeated using each RCRB associated with the function under test, or associated with an RCRB that is itself associated with the function under test. See Section 2.1.2.17 for details. All offsets in the tests are now relative to the RCRB's base address and all cycles are now memory space accesses.

The test *fails* if:

- ☐ More than one MR-IOV capability structure is present.
- ☐ An RCRB contains a MR-IOV capability structure.
- ☐ The Capability Version field does not report 1h.
- ☐ The Next Capability Offset field does not report 000h or a value greater than 0FFh or the lower two bits are not 00b.
- ☐ The MR-IOV capability structure is present in a device that is not an Endpoint or a Switch.
- ☐ The MR-IOV capability structure is present in a VF.

2.2.53. TD_1_58 Secondary PCI Express Extended Capability Structure

The test verifies that if the function under test reports a Secondary PCI Express Extended Capability structure, it is implemented as defined in the relevant specifications.

The test will only run for Base 3.x or later testing.

Relevant Specifications

□ *PCI Express Base Specification*

(See Section 1.3 for additional specification revision details.)

Applicable Device/Port Types

This test is run on all device/port types.

This test is run on any RCRB associated with the function under test.

Starting Configuration

This test is run with functions starting in the D0-Uninitialized state following the standard initialization procedures in Section 2.1.1. Testing in additional Dx states may be optionally supported. ASPM states of the upstream and downstream ports are disabled for this test.

Overview of Test Steps

Test software performs the following steps:

1. Configure the function under test following the procedure described in Section 2.1.1.
2. Examine the Extended Capability ID field for each of the detected Extended Capabilities. Determine how many instances of the Extended Capability ID of 0019h (Secondary PCI Express Extended Capability) are found. If more than one is found, the test terminates with a failure.
3. For Base 2.x or earlier testing: if the Extended Capability ID of 0019h is found for an extended capability, the test terminates with a failure.
4. For Base 3.x or later testing: if this is an RCRB, then the following tests are performed:
 - a. If the RCRB does not contain a Root Complex Internal Link Control extended capability (Extended Capability ID of 0006h), then if the Extended Capability ID of 0019h is found for an extended capability, the test terminates with a failure.
 - b. If the RCRB contains a Root Complex Internal Link Control extended capability (Extended Capability ID of 0006h) then:
 - i. If the RCRB under test reports the Max Link Speed field (Root Complex Link Capabilities register) of 0011b or greater or the Supported Link Speeds Vector field (Root Complex Link Capabilities register) of 100b or greater, then if the Extended Capability ID of 0019h is not found for an extended capability, the test terminates with a failure.
5. For Base 3.x or later testing: if this is not an RCRB, then the following tests are performed:
 - a. If the device type is a Root Complex Integrated Endpoint or a Root Complex Event Collector, then if the Extended Capability ID of 0019h is found for an extended capability, the test terminates with a failure.

- b. If the device type is not a Root Complex Integrated Endpoint or a Root Complex Event Collector, and if this is not function 0, then if the Extended Capability ID of 0019h is found for an extended capability, the test terminates with a failure.
 - c. If the device type is not a Root Complex Integrated Endpoint or a Root Complex Event Collector, and if this is function 0 then:
 - i. If the function under test reports the Max Link Speed field (Link Capabilities register) of 0011b or greater, then if the Extended Capability ID of 0019h is not found for an extended capability, the test terminates with a failure.
 - ii. If the function under test reports the Capability Version field (PCI Express Capabilities register) is 2h or greater:
 - a. If the function under test reports the Supported Link Speeds Vector field (Link Capabilities 2 register) of 100b or greater, then if the Extended Capability ID of 0019h is not found for an extended capability, the test terminates with a failure.
6. For Base 3.x or later testing: if an Extended Capability ID of 0019h is found for an extended capability the following checks are performed on that extended capability structure:
- a. The Capability Version field must be 1h.
 - b. The Next Capability Offset field must be 000h or greater than 0FFh and the lower 2 bits of this field must be 00b.

7. The following register field characteristic checks are performed:

Secondary PCI Express Extended Capability Header (Offset 00h) — DWORD

- a. PCI Express Extended Capability ID RO
- b. Capability Version RO
- c. Next Capability Offset RO

Link Control 3 Register (Offset 04h) — DWORD

- a. Perform Equalization
 - (for Upstream Ports with Crosslink Supported as 0) RO-Zero
 - (for Upstream Ports with Crosslink Supported as 1) RW
 - (for Downstream Ports) RW

Note: This test must not write 1 to the Retrain Link field (Link Control register), when this field is 1. Also, the test must restore this field to 0 when the test completes.
- b. Link Equalization Request Interrupt Enable
 - (for Upstream Ports with Crosslink Supported as 0) RO-Zero
 - (for Upstream Ports with Crosslink Supported as 1) RW
 - (for Downstream Ports) RW
- c. RsvdP_31-2 RO-Zero

Lane Error Status Register (Offset 08h) — DWORD

- a. (bits [Maximum Link Width-1] - 0) RW1CS
- b. (if [Maximum Link Width] is less than 32:
 - bits 31 - [Maximum Link Width]) RO-Zero

8. The following default value checks are performed:

Link Control 3 Register Default Value (Offset 04h) — DWORD

(all except Upstream Ports with Crosslink Supported as 0)

- a. Perform Equalization 0

Note: This test must not write 1 to the Retrain Link field (Link Control register), when this field is 1.
- b. Link Equalization Request Interrupt Enable 0

9. For each Equalization Control register set value [n] (given by value in Maximum Link Width - 1), the following register field characteristic checks are performed:

Equalization Control Register (n) (Offset 0Ch + n * 02h) – WORD

- | | | |
|----|---|---------|
| a. | Downstream Port Transmitter Preset | |
| | (for Upstream Ports with Crosslink Supported as 0) | RO-Zero |
| | (for Upstream Ports with Crosslink Supported as 1, for non-FLR testing) | HwInit |
| | (for Upstream Ports with Crosslink Supported as 1, for FLR testing) | RO |
| | (for Downstream Ports, for non-FLR testing) | HwInit |
| | (for Downstream Ports, for FLR testing) | RO |
| b. | Downstream Port Receiver Preset Hint | |
| | (for Upstream Ports with Crosslink Supported as 0) | RO-Zero |
| | (for Upstream Ports with Crosslink Supported as 1, for non-FLR testing) | HwInit |
| | (for Upstream Ports with Crosslink Supported as 1, for FLR testing) | RO |
| | (for Downstream Ports, for non-FLR testing) | HwInit |
| | (for Downstream Ports, for FLR testing) | RO |
| c. | RsvdP_7 | RO-Zero |
| d. | Upstream Port Transmitter Preset | |
| | (for Upstream Ports with Crosslink Supported as 0) | RO |
| | (for Upstream Ports with Crosslink Supported as 1, for non-FLR testing) | HwInit |
| | (for Upstream Ports with Crosslink Supported as 1, for FLR testing) | RO |
| | (for Downstream Ports, for non-FLR testing) | HwInit |
| | (for Downstream Ports, for FLR testing) | RO |
| e. | Upstream Port Receiver Preset Hint | |
| | (for Upstream Ports with Crosslink Supported as 0) | RO |
| | (for Upstream Ports with Crosslink Supported as 1, for non-FLR testing) | HwInit |
| | (for Upstream Ports with Crosslink Supported as 1, for FLR testing) | RO |
| | (for Downstream Ports, for non-FLR testing) | HwInit |
| | (for Downstream Ports, for FLR testing) | RO |
| f. | RsvdP_15 | RO-Zero |
10. For functions under test that have a link, the test is run at each of the following link speeds:
- 2.5 GT/s (for devices that are capable). See Section 2.1.2.12 for details.
 - 5.0 GT/s (for devices that are capable). See Section 2.1.2.13 for details.
 - 8.0 GT/s (for devices that are capable). See Section 2.1.2.14 for details.
11. If the device type is Root Port or Root Complex Integrated Endpoint, then the test is repeated using each RCRB associated with the function under test, or associated with an RCRB that is itself associated with the function under test. See Section 2.1.2.17 for details. All offsets in the tests are now relative to the RCRB's base address and all cycles are now memory space accesses.

The test *fails* if:

- ☐ More than one Secondary PCI Express Capability is present.
- ☐ An RCRB that does not correspond to a Root Complex Internal Link and the Secondary PCI Express extended capability structure is present.
- ☐ An RCRB that corresponds to a Root Complex Internal Link and that supports 8.0 GT/s, and the Secondary PCI Express extended capability structure is not present.

- ☐ A Root Complex Integrated Endpoint or a Root Complex Event Collector, and the Secondary PCI Express extended capability structure is present.
- ☐ A function other than 0, and the Secondary PCI Express extended capability structure is present.
- ☐ A device type that supports 8.0 GT/s, and the Secondary PCI Express extended capability structure is not present.
- ☐ The Capability Version field does not report 1h.
- ☐ The Next Capability Offset field does not report 000h or a value greater than 0FFh or the lower two bits are not 00b.
- ☐ Any of the register field characteristic tests fail.
- ☐ Any of the default value tests fail.

2.2.54. TD_1_59 Protocol Multiplexing Extended Capability Structure

The test verifies that if the function under test reports a Protocol Multiplexing (PMUX) Extended Capability structure, it is implemented as defined in the relevant specifications.

The test will only run for Base 3.x or later testing.

Relevant Specifications

- ☐ *PCI Express Base Specification*

(See Section 1.3 for additional specification revision details.)

Applicable Device/Port Types

This test is run on all device/port types.

This test is run on any RCRB associated with the function under test.

Starting Configuration

This test is run with functions starting in the D0-Uninitialized state following the standard initialization procedures in Section 2.1.1. Testing in additional Dx states may be optionally supported. ASPM states of the upstream and downstream ports are disabled for this test.

Overview of Test Steps

Test software performs the following steps:

1. Configure the function under test following the procedure described in Section 2.1.1.
2. Examine the Extended Capability ID field for each of the detected Extended Capabilities. Determine how many instances of the Extended Capability ID of 001Ah (Protocol Multiplexing Extended Capability) are found. If more than one is found, the test terminates with a failure.
3. For Base 2.x or earlier testing: if the Extended Capability ID of 001Ah is found for an extended capability, the test terminates with a failure.
4. If the Extended Capability ID of 001Ah is found in an RCRB, the test terminates with a failure.

5. For Base 3.x or later testing: if this is not an RCRB, then the following tests are performed:
 - a. If the device type is a Root Complex Integrated Endpoint or a Root Complex Event Collector, then if the Extended Capability ID of 001Ah is found for an extended capability, the test terminates with a failure.
 - b. If the device type is not a Root Complex Integrated Endpoint or a Root Complex Event Collector, and if this is not function 0, then if the Extended Capability ID of 001Ah is found for an extended capability, the test terminates with a failure.
 - c. If the device type is not a Root Complex Integrated Endpoint or a Root Complex Event Collector, and if this is function 0 and the Capability Version field (PCI Express Capabilities register) is 1h or less, then if the Extended Capability ID of 001Ah is found for an extended capability, the test terminates with a failure.
6. For Base 3.x or later testing: if an Extended Capability ID of 001Ah is found for an extended capability, the following checks are performed on that extended capability structure:
 - a. The Capability Version field must be 1h.
 - b. The Next Capability Offset field must be 000h or greater than 0FFh and the lower 2 bits of this field must be 00b.
 - c. For Base 3.x testing: the PMUX Supported Link Speeds field (PMUX Capability register) must be 0000 0001b, 0000 0010b, 0000 0011b, 0000 0100b, 0000 0101b, 0000 0110b, or 0000 0111b. All other encodings are reserved and treated as a failure.
 - d. If the Supported Link Speeds Vector field (Link Capabilities 2 register) bit 0 (2.5 GT/s) returns 0, then the PMUX Supported Link Speeds field (PMUX Capability register) bit 0 (2.5 GT/s) must be 0.
 - e. If the Supported Link Speeds Vector field (Link Capabilities 2 register) bit 1 (5.0 GT/s) returns 0, then the PMUX Supported Link Speeds field (PMUX Capability register) bit 1 (5.0 GT/s) must be 0.
 - f. If the Supported Link Speeds Vector field (Link Capabilities 2 register) bit 2 (8.0 GT/s) returns 0, then the PMUX Supported Link Speeds field (PMUX Capability register) bit 2 (8.0 GT/s) must be 0.
7. The following register field characteristic checks are performed:

PMUX Extended Capability Header (Offset 00h) — DWORD

- | | |
|---------------------------------------|----|
| a. PCI Express Extended Capability ID | RO |
| b. Capability Version | RO |
| c. Next Capability Offset | RO |

PMUX Capability Register (Offset 04h) — DWORD

- | | |
|-------------------------------|---------|
| a. PMUX Protocol Array Size | RO |
| b. RsvdP_7-6 | RO-Zero |
| c. PMUX Supported Link Speeds | RO |
| d. RsvdP_31-16 | RO-Zero |

PMUX Control Register (Offset 08h) — DWORD

- | | |
|--|------------------|
| a. PMUX Channel 0 Assignment | |
| (for PMUX Protocol Array Size equal to 0) | |
| (bits 5-0) | RW or
RO-Zero |
| (for PMUX Protocol Array Size equal to 1) | |
| (bit 0) | RW |
| (bits 5-1) | RW or
RO-Zero |
| (for PMUX Protocol Array Size equal to or between 3 and 2) | |
| (bits 1-0) | RW |
| (bits 5-2) | RW or
RO-Zero |
| (for PMUX Protocol Array Size equal to or between 7 and 4) | |
| (bits 2-0) | RW |
| (bits 5-3) | RW or
RO-Zero |
| (for PMUX Protocol Array Size equal to or between 15 and 8) | |
| (bits 3-0) | RW |
| (bits 5-4) | RW or
RO-Zero |
| (for PMUX Protocol Array Size equal to or between 31 and 16) | |
| (bits 4-0) | RW |
| (bit 5) | RW or
RO-Zero |
| (for PMUX Protocol Array Size greater than 31) | |
| (bits 5-0) | RW
RO-Zero |
| b. RsvdP_7-6 | |
| c. PMUX Channel 1 Assignment | |
| (for PMUX Protocol Array Size equal to 0) | |
| (bits 5-0) | RW or
RO-Zero |
| (for PMUX Protocol Array Size equal to 1) | |
| (bit 0) | RW |
| (bits 5-1) | RW or
RO-Zero |
| (for PMUX Protocol Array Size equal to or between 3 and 2) | |
| (bits 1-0) | RW |
| (bits 5-2) | RW or
RO-Zero |
| (for PMUX Protocol Array Size equal to or between 7 and 4) | |
| (bits 2-0) | RW |
| (bits 5-3) | RW or
RO-Zero |

	(for PMUX Protocol Array Size equal to or between 15 and 8)	RW
	(bits 3-0)	RW or
	(bits 5-4)	RO-Zero
	(for PMUX Protocol Array Size equal to or between 31 and 16)	RW
	(bits 4-0)	RW or
	(bit 5)	RO-Zero
	(for PMUX Protocol Array Size greater than 31)	RW
	(bits 5-0)	RO-Zero
d.	RsvdP_15-14	
e.	PMUX Channel 2 Assignment	
	(for PMUX Protocol Array Size equal to 0)	
	(bits 5-0)	RW or
		RO-Zero
	(for PMUX Protocol Array Size equal to 1)	
	(bit 0)	RW
	(bits 5-1)	RW or
		RO-Zero
	(for PMUX Protocol Array Size equal to or between 3 and 2)	
	(bits 1-0)	RW
	(bits 5-2)	RW or
		RO-Zero
	(for PMUX Protocol Array Size equal to or between 7 and 4)	
	(bits 2-0)	RW
	(bits 5-3)	RW or
		RO-Zero
	(for PMUX Protocol Array Size equal to or between 15 and 8)	
	(bits 3-0)	RW
	(bits 5-4)	RW or
		RO-Zero
	(for PMUX Protocol Array Size equal to or between 31 and 16)	
	(bits 4-0)	RW
	(bit 5)	RW or
		RO-Zero
	(for PMUX Protocol Array Size greater than 31)	
	(bits 5-0)	RW
f.	RsvdP_23-22	RO-Zero
g.	PMUX Channel 3 Assignment	
	(for PMUX Protocol Array Size equal to 0)	
	(bits 5-0)	RW or
		RO-Zero
	(for PMUX Protocol Array Size equal to 1)	
	(bit 0)	RW
	(bits 5-1)	RW or
		RO-Zero

(for PMUX Protocol Array Size equal to or between 3 and 2)
(bits 1-0)
(bits 5-2)

RW
RW or
RO-Zero

(for PMUX Protocol Array Size equal to or between 7 and 4)
(bits 2-0)
(bits 5-3)

RW
RW or
RO-Zero

(for PMUX Protocol Array Size equal to or between 15 and 8)
(bits 3-0)
(bits 5-4)

RW
RW or
RO-Zero

(for PMUX Protocol Array Size equal to or between 31 and 16)
(bits 4-0)
(bit 5)

RW
RW or
RO-Zero

(for PMUX Protocol Array Size greater than 31)
(bits 5-0)

RW
RO-Zero

h. RsvdP_31-30

PMUX Status Register (Offset 0Ch) — DWORD

- a. PMUX Channel 0 Disabled: Link Speed
- b. PMUX Channel 0 Disabled: Link Width
- c. PMUX Channel 0 Disabled: Protocol Specific
- d. RsvdZ_7-3
- e. PMUX Channel 1 Disabled: Link Speed
- f. PMUX Channel 1 Disabled: Link Width
- g. PMUX Channel 1 Disabled: Protocol Specific
- h. RsvdZ_15-11
- i. PMUX Channel 2 Disabled: Link Speed
- j. PMUX Channel 2 Disabled: Link Width
- k. PMUX Channel 2 Disabled: Protocol Specific
- l. RsvdZ_23-19
- m. PMUX Channel 3 Disabled: Link Speed
- n. PMUX Channel 3 Disabled: Link Width
- o. PMUX Channel 3 Disabled: Protocol Specific
- p. RsvdZ_31-27

RO
RO
RO
RO-Zero
RO
RO
RO
RO-Zero
RO
RO
RO
RO-Zero
RO
RO
RO
RO-Zero

8. The following default value checks are performed:

PMUX Control Register Default Value (Offset 08h) — DWORD

- a. PMUX Channel 0 Assignment
- b. PMUX Channel 1 Assignment
- c. PMUX Channel 2 Assignment
- d. PMUX Channel 3 Assignment

00 0000b
00 0000b
00 0000b
00 0000b

PMUX Status Register Default Value (Offset 0Ch) — DWORD

a. PMUX Channel 0 Disabled: Link Speed	0
b. PMUX Channel 0 Disabled: Link Width	0
c. PMUX Channel 0 Disabled: Protocol Specific	0
d. PMUX Channel 1 Disabled: Link Speed	0
e. PMUX Channel 1 Disabled: Link Width	0
f. PMUX Channel 1 Disabled: Protocol Specific	0
g. PMUX Channel 2 Disabled: Link Speed	0
h. PMUX Channel 2 Disabled: Link Width	0
i. PMUX Channel 2 Disabled: Protocol Specific	0
j. PMUX Channel 3 Disabled: Link Speed	0
k. PMUX Channel 3 Disabled: Link Width	0
l. PMUX Channel 3 Disabled: Protocol Specific	0

Note: PMUX Status bits for a channel must all return 0 when that channel's Control assignment bit is 0. Also, each channels Control assignment bit is required to default to 0.

9. For each PMUX Protocol Array Entry register set value [n] (given by value in PMUX Protocol Array Size - 1), the following register field characteristic checks are performed:

PMUX Protocol Array Entry (n + 1) (Offset 10h + n * 04h) – DWORD

a. Protocol ID	RO
b. Authority ID	RO

10. For functions under test that have a link, the test is run at each of the following link speeds:
- a. 2.5 GT/s (for devices that are capable). See Section 2.1.2.12 for details.
 - b. 5.0 GT/s (for devices that are capable). See Section 2.1.2.13 for details.
 - c. 8.0 GT/s (for devices that are capable). See Section 2.1.2.14 for details.
11. If the device type is Root Port or Root Complex Integrated Endpoint, then the test is repeated using each RCRB associated with the function under test, or associated with an RCRB that is itself associated with the function under test. See Section 2.1.2.17 for details. All offsets in the tests are now relative to the RCRB's base address and all cycles are now memory space accesses.

The test *fails* if:

- ☐ More than one Protocol Multiplexing Capability is present.
- ☐ An RCRB contains a Protocol Multiplexing capability structure.
- ☐ A Root Complex Integrated Endpoint or a Root Complex Event Collector, and the Protocol Multiplexing extended capability structure is present.
- ☐ A function other than 0, and the Protocol Multiplexing extended capability structure is present.
- ☐ The Capability Version field does not report 1h.
- ☐ The Next Capability Offset field does not report 000h or a value greater than 0FFh or the lower two bits are not 00b.
- ☐ The PMUX Supported Link Speeds field is not one of the defined values.
- ☐ The PMUX Supported Link Speeds field reports support for a link speed that is not supported in the value reported in the Supported Link Speeds Vector field.
- ☐ Any of the register field characteristic tests fail.
- ☐ Any of the default value tests fail.

2.3. Configuration Register Tests (Devices with Upstream Ports)

This set of tests applies only to upstream ports of Switches, Bridges, or Endpoint devices.

2.3.1. TD_2_1 ASPM Configuration Stress (Upstream Ports)

The test verifies that if the function under test implements ASPM, test software attempts to configure the function under test to a variety of different ASPM settings will not cause the device to operate abnormally, as defined in the relevant specifications.

Relevant Specifications

□ *PCI Express Base Specification*

(See Section 1.3 for additional specification revision details.)

Applicable Device/Port Types

This test is run on upstream ports of PCI Express Endpoints, Legacy Endpoints, Switch Upstream Ports, and PCI Express to PCI/PCI-X Bridges.

(Note: The downstream port connected to the function under test must be known to operate correctly in each of its supported ASPM modes.)

Starting Configuration

Function under test is completely uninitialized following a reset.

Overview of Test Steps

Test software performs the following steps:

1. Configure the function under test following the procedure described in Section 2.1.1. If the procedure fails to complete successfully, the test terminates with failure.
2. Read the Active State Power Management (ASPM) Support field from the Link Capabilities register of the function under test (this will also be referred to as the upstream port of the link). Record this value [UASPM].
3. Read the Active State Power Management (ASPM) Support field from the Link Capabilities register of the immediate upstream device's port (the downstream port of the link connected to the function under test), which will be referred to as the downstream port of the link. Record this value [DASPM].
4. Set ASPM disabled (Active State Power Management (ASPM) Control = 00b in Link Control register) for both the function under test and the downstream port of the link. If the function under test is a VF, then the ASPM setting value must be written to the PF associated with that VF. Repeat step 1 for 100 iterations.
5. Use the level of testing to determine the interpretation of the Active State Power Management (ASPM) Support field, and what values to write to the Active State Power Management (ASPM) Control field in the Link Control register of the upstream port of the link and the downstream port of the link. If the function under test is not a VF, but is part of a multi-function device (Header Type bit 7 is 1), then the same ASPM setting value must be written to all non-VF functions in that multi-function device. If the function under test is a VF, then the ASPM

setting value must be written to the PF associated with that VF, as well as any other non-VF functions in the multi-function device.

- a. For Base 1.x testing: use Table 1 in Section 2.1.2.19.1, together with [UASPM] and [DASPM] to determine which ASPM combinations are supported, repeat step 1 for 100 iterations for each of the supported ASPM combinations in Table 1 in Section 2.1.2.19.1.
- b. For Base 2.x or later testing: read the ASPM Optionality Compliance field from the Link Capabilities register in both the upstream port of the link and the downstream port of the link, and based on their reported values perform the following:
 - i. If the ASPM Optionality Compliance field is 0 in both the upstream port of the link and the downstream port of the link, use Table 1 in Section 2.1.2.19.1, together with [UASPM] and [DASPM] to determine which ASPM combinations are supported, repeat step 1 for 100 iterations for each of the supported ASPM combinations in Table 1 in Section 2.1.2.19.1.
 - ii. If the ASPM Optionality Compliance field is 1 in the upstream port of the link and the ASPM Optionality Compliance field is 0 in the downstream port of the link, use Table 2 in Section 2.1.2.19.2, together with [UASPM] and [DASPM] to determine which ASPM combinations are supported, repeat step 1 for 100 iterations for each of the supported ASPM combinations in Table 2 in Section 2.1.2.19.2.
 - iii. If the ASPM Optionality Compliance field is 0 in the upstream port of the link and the ASPM Optionality Compliance field is 1 in the downstream port of the link, use Table 3 in Section 2.1.2.19.3, together with [UASPM] and [DASPM] to determine which ASPM combinations are supported, repeat step 1 for 100 iterations for each of the supported ASPM combinations in Table 3 in Section 2.1.2.19.3.
 - iv. If the ASPM Optionality Compliance field is 1 in both the upstream port of the link and the downstream port of the link, use Table 4 in Section 2.1.2.19.4, together with [UASPM] and [DASPM] to determine which ASPM combinations are supported, repeat step 1 for 100 iterations for each of the supported ASPM combinations in Table 4 in Section 2.1.2.19.4.
6. For functions under test that have a link, the test is run at each of the following link speeds:
 - a. 2.5 GT/s (for devices that are capable). See Section 2.1.2.12 for details.
 - b. 5.0 GT/s (for devices that are capable). See Section 2.1.2.13 for details.
 - c. 8.0 GT/s (for devices that are capable). See Section 2.1.2.14 for details.

The test *fails* if:

- ☐ The function under test fails to respond to any part of the standard configuration sequence.

2.3.2. TD_2_2 Link Training Stress

The test verifies that the function under test correctly handles link retraining when it is initiated through the Link Control register of the downstream port connected to the function under test as required in the relevant specifications.

Relevant Specifications

- ☐ *PCI Express Base Specification*

(See Section 1.3 for additional specification revision details.)

Applicable Device/Port Types

This test is run on upstream ports of PCI Express Endpoints, Legacy Endpoints, Switch Upstream Ports, and PCI Express to PCI/PCI-X Bridges.

(Note: The downstream port connected to the function under test must be known to operate correctly in each of its supported ASPM modes.)

Starting Configuration

The device is placed into the desired starting state (D0-Uninitialized) following the standard initialization sequence in Section 2.1.1. Testing in additional Dx states may be optionally supported.

Overview of Test Steps

Test software performs the following steps:

1. Test software resets the bus connected to the function under test following the procedure describe in Section 2.1.1.1.
2. Read the Active State Power Management (ASPM) Support field from the Link Capabilities register of the function under test (this will also be referred to as the upstream port of the link). Record this value [UASPM].
3. Read the Active State Power Management (ASPM) Support field from the Link Capabilities register of the immediate upstream device's port (the downstream port of the link connected to the function under test), which will be referred to as the downstream port of the link. Record this value [DASPM].
4. Test software configures the function under test following the procedure described in Section 2.1.1. If the procedure fails to complete successfully, the test terminates with failure.
5. Test software modifies at least one register field value from its default value.
6. Test software reads the Negotiated Link Width field of the Link Status register for both the function under test and the downstream port of the link. Both values must be the same otherwise the test terminates with failure. If the same, the value is recorded in the link width value [LWV].
7. Test software writes 1 to the Retrain Link field (Link Control register) on the downstream port of the link using a WORD access, while preserving all the other fields in this register.
8. Test software reads the Link Training field (Link Status register) on the downstream port of the link, until it returns 0.
9. Test software reads the Negotiated Link Width field (Link Status register) for both the function under test and the downstream port of the link. Both values must be the same otherwise the test terminates with failure. If the same, the value must be the same as [LWV] otherwise the test terminates with failure.
10. Test software reads back the register fields programmed to a non-default value in step 5 and confirms it has not changed back to its default value.
11. Steps 4-11 are repeated for 100 iterations.

12. Test software repeats steps 1-12 with different combinations of ASPM settings. Test software uses the level of testing to determine the interpretation of the Active State Power Management (ASPM) Support field, and what values to write to the Active State Power Management (ASPM) Control field in the Link Control register of the upstream port of the link and the downstream port of the link. If the function under test is not a VF, but is part of a multi-function device (Header Type bit 7 is 1), then the same ASPM setting value must be written to all non-VF functions in that multi-function device. If the function under test is a VF, then the ASPM setting value must be written to the PF associated with that VF, as well as any other non-VF functions in the multi-function device.
 - a. For Base 1.x testing: use Table 1 in Section 2.1.2.19.1, together with [UASPM] and [DASPM] to determine which ASPM combinations are supported, repeats step 4-10 for 100 iterations for each of the supported ASPM combinations in Table 1 in Section 2.1.2.19.1.
 - b. For Base 2.x or later testing: read the ASPM Optionality Compliance field from the Link Capabilities register in both the upstream port of the link and the downstream port of the link, and based on their reported values perform the following:
 - i. If the ASPM Optionality Compliance field is 0 in both the upstream port of the link and the downstream port of the link, use Table 1 in Section 2.1.2.19.1, together with [UASPM] and [DASPM] to determine which ASPM combinations are supported, repeat steps 4-10 for 100 iterations for each of the supported ASPM combinations in Table 1 in Section 2.1.2.19.1.
 - ii. If the ASPM Optionality Compliance field is 1 in the upstream port of the link and the ASPM Optionality Compliance field is 0 in the downstream port of the link, use Table 2 in Section 2.1.2.19.2, together with [UASPM] and [DASPM] to determine which ASPM combinations are supported, repeat steps 4-10 for 100 iterations for each of the supported ASPM combinations in Table 2 in Section 2.1.2.19.2.
 - iii. If the ASPM Optionality Compliance field is 0 in the upstream port of the link and the ASPM Optionality Compliance field is 1 in the downstream port of the link, use Table 3 in Section 2.1.2.19.3, together with [UASPM] and [DASPM] to determine which ASPM combinations are supported, repeat steps 4-10 for 100 iterations for each of the supported ASPM combinations in Table 3 in Section 2.1.2.19.3.
 - iv. If the ASPM Optionality Compliance field is 1 in both the upstream port of the link and the downstream port of the link, use Table 4 in Section 2.1.2.19.4, together with [UASPM] and [DASPM] to determine which ASPM combinations are supported, repeat steps 4-10 for 100 iterations for each of the supported ASPM combinations in Table 4 in Section 2.1.2.19.4.
13. For functions under test that have a link, the test is run at each of the following link speeds:
 - a. 2.5 GT/s (for devices that are capable). See Section 2.1.2.12 for details.
 - b. 5.0 GT/s (for devices that are capable). See Section 2.1.2.13 for details.
 - c. 8.0 GT/s (for devices that are capable). See Section 2.1.2.14 for details.

The test *fails* if:

- ☐ The reported Negotiated Link Width of the upstream component ever fails to match the Negotiated Link Width of the downstream component.
- ☐ Link retraining causes the Negotiated Link Width to change.
- ☐ Link retraining causes the upstream port to reset (restoring default values).

2.3.3. TD_2_3 Tolerance of Hot-Plug Signaling/Ignored Messages

The test verifies that the function under test continues to be configurable and respond normally when Hot-Plug Signaling/Ignored messages are intermixed with the standard configuration sequence as required in the relevant specifications.

Relevant Specifications

□ *PCI Express Base Specification*

(See Section 1.3 for additional specification revision details.)

Applicable Device/Port Types

This test is run on upstream ports of PCI Express Endpoints, Legacy Endpoints, Switch Upstream Ports, and PCI Express to PCI/PCI-X Bridges.

(Note: The downstream port originating the link hierarchy containing the function under test must be known to generate Hot-Plug Signaling message types. As this support was deleted in any Base 1.1 compliant device, these message types are now defined as Ignored messages and are not to be generated by any Base 1.1 or later compliant port.)

Starting Configuration

The device is placed into the desired starting state (D0-Uninitialized) following the standard initialization sequence in Section 2.1.1. Testing in additional Dx states may be optionally supported.

Overview of Test Steps

Test software performs the following steps:

1. Check that the downstream port originating the link hierarchy containing the function under test is capable of sending Hot-Plug Messages (Attention Indicator, or Power Indicator). If not skip this test (this is not a failure).
2. Configure the function under test following the procedure described in Section 2.1.1 (D0-Initialized).
3. At the end of the configuration sequence test software writes 01b to the Attention Indicator Control field of the downstream port to which the function under test is attached. This causes an ATTENTION_INDICATOR message to be sent to the function under test.
4. Test software reads standard configuration registers to ensure that the function under test is still responding normally.
5. The standard configuration sequence described in Section 2.1.1 (D0-Initialized) is run with a random Attention or Power Indicator control message sent between every normal configuration access.
6. Step 5 is repeated for 100 iterations.
7. A flood of Attention/Power Indicator control messages (500) are sent to the function under test.
8. Test software ensures the device configuration registers can still be accessed normally.

9. Note: If the function under test has Attention and Power Indicators present on the card, test software uses other mechanisms to send unsupported messages to the function under test (if possible).
10. For functions under test that have a link, the test is run at each of the following link speeds:
 - a. 2.5 GT/s (for devices that are capable). See Section 2.1.2.12 for details.
 - b. 5.0 GT/s (for devices that are capable). See Section 2.1.2.13 for details.
 - c. 8.0 GT/s (for devices that are capable). See Section 2.1.2.14 for details.

The test *fails* if:

- ☐ The function under test fails to respond normally to configuration accesses at any time.
- ☐ The function under test produces an error in response to an Attention or Power Indicator control message.

2.3.4. TD_2_4 Response To Earliest Allowed Configuration Requests After Reset

The test verifies that the function under test operates normally when test software attempts to configure the function under test such that configuration requests begin as early as allowed after reset as required in the relevant specifications.

Relevant Specifications

- ☐ *PCI Express Base Specification*

(See Section 1.3 for additional specification revision details.)

Applicable Device/Port Types

This test is run on upstream ports of PCI Express Endpoints, Legacy Endpoints, Switch Upstream Ports, and PCI Express to PCI/PCI-X Bridges.

(Note: The downstream port connected to the function under test must be known to operate correctly in each of its supported ASPM modes.)

Starting Configuration

Function under test is completely uninitialized following a reset.

Overview of Test Steps

Test software performs the following steps:

1. Configure the function under test following the procedure described in Section 2.1.1. In all cases the first configuration request following reset is sent at the earliest allowed time (100 ms). If the procedure fails to complete successfully, the test terminates with failure.
2. Set ASPM disabled (Active State Power Management (ASPM) Control = 00b in Link Control register) for both the function under test and the downstream port of the link. If the function under test is a VF, then the ASPM setting value must be written to the PF associated with that VF. Repeat step 1 for 100 iterations.
3. Read the Active State Power Management (ASPM) Support field from the Link Capabilities register of the function under test (this will also be referred to as the upstream port of the link). Record this value [UASPM].

4. Read the Active State Power Management (ASPM) Support field from the Link Capabilities register of the immediate upstream device's port (the downstream port of the link connected to the function under test), which will be referred to as the downstream port of the link. Record this value [DASPM].
5. Use the level of testing to determine the interpretation of the Active State Power Management (ASPM) Support field, and what values to write to the Active State Power Management (ASPM) Control field in the Link Control register of the upstream port of the link and the downstream port of the link, set immediately after bus reset. If the function under test is not a VF, but is part of a multi-function device (Header Type bit 7 is 1), then the same ASPM setting value must be written to all non-VF functions in that multi-function device. If the function under test is a VF, then the ASPM setting value must be written to the PF associated with that VF, as well as any other non-VF functions in the multi-function device.
 - a. For Base 1.x testing: use Table 1 in Section 2.1.2.19.1, together with [UASPM] and [DASPM] to determine which ASPM combinations are supported, repeat step 1 for 100 iterations for each of the supported ASPM combinations in Table 1 in Section 2.1.2.19.1.
 - b. For Base 2.x or later testing: read the ASPM Optionality Compliance field from the Link Capabilities register in both the upstream port of the link and the downstream port of the link, and based on their reported values perform the following:
 - i. If the ASPM Optionality Compliance field is 0 in both the upstream port of the link and the downstream port of the link, use Table 1 in Section 2.1.2.19.1, together with [UASPM] and [DASPM] to determine which ASPM combinations are supported, repeat step 1 for 100 iterations for each of the supported ASPM combinations in Table 1 in Section 2.1.2.19.1.
 - ii. If the ASPM Optionality Compliance field is 1 in the upstream port of the link and the ASPM Optionality Compliance field is 0 in the downstream port of the link, use Table 2 in Section 2.1.2.19.2, together with [UASPM] and [DASPM] to determine which ASPM combinations are supported, repeat step 1 for 100 iterations for each of the supported ASPM combinations in Table 2 in Section 2.1.2.19.2.
 - iii. If the ASPM Optionality Compliance field is 0 in the upstream port of the link and the ASPM Optionality Compliance field is 1 in the downstream port of the link, use Table 3 in Section 2.1.2.19.3, together with [UASPM] and [DASPM] to determine which ASPM combinations are supported, repeat step 1 for 100 iterations for each of the supported ASPM combinations in Table 3 in Section 2.1.2.19.3.
 - iv. If the ASPM Optionality Compliance field is 1 in both the upstream port of the link and the downstream port of the link, use Table 4 in Section 2.1.2.19.4, together with [UASPM] and [DASPM] to determine which ASPM combinations are supported, repeat step 1 for 100 iterations for each of the supported ASPM combinations in Table 4 in Section 2.1.2.19.4.
6. For functions under test that have a link, the test is run at each of the following link speeds:
 - a. 2.5 GT/s (for devices that are capable). See Section 2.1.2.12 for details.
 - b. 5.0 GT/s (for devices that are capable). See Section 2.1.2.13 for details.
 - c. 8.0 GT/s (for devices that are capable). See Section 2.1.2.14 for details.

The test *fails* if:

- ☐ The function under test fails to respond to any part of the standard configuration sequence.

2.3.5. TD_2_5 Response to Different Bus and Device Numbers

The test verifies that the function under test views itself as the recipient of any Type 0 configuration requests it receives regardless of the Bus Number in the configuration request. Additionally, for a non-ARI device it is required to view itself as the recipient of any Type 0 configuration requests it receives regardless of the Device Number in the configuration request. This is defined in the relevant specifications.

Relevant Specifications

☐ *PCI Express Base Specification*

(See Section 1.3 for additional specification revision details.)

Applicable Device/Port Types

This test is run on upstream ports of PCI Express Endpoints, Legacy Endpoints, Switch Upstream Ports, and PCI Express to PCI/PCI-X Bridges.

Starting Configuration

Function under test is completely uninitialized following a reset.

Overview of Test Steps

Test software performs the following steps:

1. Configure the function under test following the procedure described in Section 2.1.1.
2. Check if the function implements the ARI Extended Capability (Extended Capability ID of 000Eh). If it does, then record that this is an ARI device, otherwise record that this is a non-ARI device.
3. If this is a non-ARI device, configure the function under test following the procedure described in Section 2.1.1, but such that at various points in the configuration sequence different device and bus numbers are used in the commands sent to the function under test. If the procedure fails to complete successfully, the test terminates with failure.
4. If this is an ARI device, configure the function under test following the procedure described in Section 2.1.1, but such that at various points in the configuration sequence different bus numbers are used in the commands sent to the function under test. If the procedure fails to complete successfully, the test terminates with failure.
5. Repeat the appropriate configuration sequence for the function under test (either step 3 or step 4) for 100 iterations.
6. For functions under test that have a link, the test is run at each of the following link speeds:
 - a. 2.5 GT/s (for devices that are capable). See Section 2.1.2.12 for details.
 - b. 5.0 GT/s (for devices that are capable). See Section 2.1.2.13 for details.
 - c. 8.0 GT/s (for devices that are capable). See Section 2.1.2.14 for details.

The test *fails* if:

- ☐ The function under test fails to respond to any part of the standard configuration sequence.

2.3.6. TD_2_6 Secondary Bus Reset (Upstream Ports)

The test verifies that the function under test implements the Secondary Bus Reset (Bridge Control register) behavior as defined in the relevant specifications.

Relevant Specifications

- ❑ *PCI Express Base Specification*
- ❑ *PCI Express to PCI/PCI-X Bridge Specification, Revision 1.0*

(See Section 1.3 for additional specification revision details.)

Applicable Device/Port Types

This test is run on upstream ports of Switch Upstream Ports, and PCI Express to PCI/PCI-X Bridges.

Starting Configuration

This test is run with functions starting in the D0-Uninitialized state following the standard initialization procedures in Section 2.1.1. Testing in additional Dx states may be optionally supported. ASPM states of the upstream and downstream ports are disabled for this test.

Overview of Test Steps

Test software performs the following steps:

1. Configure the function under test following the procedure described in Section 2.1.1.
2. Verify that a device is present on the secondary side of the Bridge or on any downstream port of the Switch.
3. Modify some non-sticky field in the device on the secondary side of the Bridge or on any downstream port of the Switch. (Note: Any of the following fields can be used as long as they are implemented as RW in the target: Memory Space Enable (Command register); I/O Space Enable (Command register), Bus Master Enable (Command register), Interrupt Disable (Command register). If none of these fields are implemented as RW in the target, then other non-sticky RW fields can be used. If the test software cannot find any non-sticky RW field in the target, then this step is skipped.)
4. A reset of the secondary interface is initiated by test software through writing a 1 to the Secondary Bus Reset field in the function under test (for a Bridge, the Bridge Control register in the Bridge; for a Switch, the Bridge Control register in the Switch Upstream Port) using a WORD access, while preserving all the other fields in this register, and then writing a 0 using a WORD access, while preserving all the other fields in this register. (See Section 2.1.1.1.2 for details of the reset algorithm.)
5. If a non-sticky RW field was found in step 3, after the necessary delay, verify the same non-sticky field (from step 3) in the device on the secondary side of the Bridge or on any downstream port of the Switch is reset to its default value. If the value has not changed to the default value, the test fails.
6. For functions under test that have a link, the test is run at each of the following link speeds:
 - a. 2.5 GT/s (for devices that are capable). See Section 2.1.2.12 for details.
 - b. 5.0 GT/s (for devices that are capable). See Section 2.1.2.13 for details.
 - c. 8.0 GT/s (for devices that are capable). See Section 2.1.2.14 for details.

The test *fails* if:

- ❑ For a Bridge, a non-sticky register field in the device on the secondary side of the Bridge fails to reset to its default settings.
- ❑ For a Switch, a non-sticky register field in the device on any downstream port of the Switch fails to reset to its default settings.

2.3.7. TD_2_7 Requested Link Speed (Upstream Ports)

The test verifies that the function under test can function at the correct link speed when test software attempts to configure the function under test with each supported link speed as defined in the relevant specifications. (The test initiates the link speed change from the immediate downstream port connected to the function under test, but only if that downstream port supports selecting that link speed using the Target Link Speed field. In order to pass, the test requires that the downstream port supports all link speeds defined in the relevant specification. In order to pass, the test requires that the function under test be able train to each link speed up to the maximum link speed supported by the function under test.)

The test will run for 100 iterations and if any single iteration fails the test will report a failure and stop. The iteration steps will be different for the different Base levels of testing. However, in all steps, any link speed change will be initiated by the downstream port. (Note: The downstream port is required to attempt a speed change any time the Retrain Link field (Link Control register) is set to 1 and the current link speed is not equal to the Target Link Speed field. However, the actual negotiated link speed will be determined by the lowest Target Link Speed field of both the downstream port and upstream port.)

- ❑ For Base 1.x testing, the test will train the link speed to 2.5 GT/s on both ports of the link, one hundred times.
- ❑ For Base 2.x testing, the test will toggle the downstream port target link speed between 2.5 GT/s and 5.0 GT/s, and after each second toggle, the test will toggle the upstream port target link speed between 5.0 GT/s and 2.5 GT/s. The sequence is: DP2.5+UP5.0=2.5; DP5.0+UP5.0=5.0; DP2.5+UP2.5=2.5; DP5.0+UP2.5=2.5 or 5.0; with the sequence repeating one hundred times.
- ❑ For Base 3.x testing, the test will step the downstream port target link speed through a six step sequence, and after each sixth step is completed, the test will toggle the upstream port target link speed between 8.0 GT/s, 5.0 GT/s, and 2.5 GT/s. The sequence is: DP2.5+UP8.0=2.5; DP5.0+UP8.0=5.0; DP8.0+UP8.0=8.0; DP5.0+UP8.0=5.0; DP2.5+UP8.0=2.5; DP8.0+UP8.0=8.0; DP2.5+UP5.0=2.5; DP5.0+UP5.0=5.0; DP8.0+UP5.0=5.0 or 8.0; DP5.0+UP5.0=5.0; DP2.5+UP5.0=2.5; DP8.0+UP5.0=5.0 or 8.0; DP2.5+UP2.5=2.5; DP5.0+UP2.5=2.5 or 5.0; DP8.0+UP2.5=2.5 or 8.0; DP5.0+UP2.5=2.5 or 5.0; DP2.5+UP2.5=2.5; DP8.0+UP2.5=2.5 or 8.0; with the sequence repeating for one hundred times.

(Note: For 8.0 GT/s, this test assumes that hardware initiated Link Equalization has already been performed by the downstream port connected to the function under test. The test does not attempt to perform software initiated Link Equalization.)

The test only runs if both the function under test and the downstream port connected to the function under test, report a PCI Express Capability Version of 2h or greater.

Relevant Specifications

□ *PCI Express Base Specification*

(See Section 1.3 for additional specification revision details.)

Applicable Device/Port Types

This test is run on upstream ports of PCI Express Endpoints, Legacy Endpoints, Switch Upstream Ports, and PCI Express to PCI/PCI-X Bridges, that report a PCI Express Capability Version of 2h or greater.

Starting Configuration

Function under test is completely uninitialized following a reset.

Overview of Test Steps

Test software performs the following steps:

1. Configure the function under test following the procedure described in Section 2.1.1.
2. Set iteration count value [ICNT] to 0, set downstream port iteration stage value [DSTAGE] to 0, and set upstream port iteration stage value [USTAGE] to 0.
3. For this test, the immediate upstream device's port (the downstream port of the link connected to the function under test) will be referred to as the downstream port of the link. Downstream port link speed value [DLSV] is used to control the link speed of the downstream port of the link. Initially set [DLSV] to 0001b.
4. For this test, Function 0 of the device containing the function under test will be referred to as the upstream port of the link. Upstream port link speed value [ULSV] is used to control the link speed of the upstream port of the link.
5. Read the Link Capabilities register bits 3-0 value from the upstream port of the link and assign it to upstream port supported speed value [USSV] and perform the following:
 - a. If [USSV] is 0000b, the test terminates with failure.
 - b. If [USSV] is 0001b, set [ULSV] to 0001b.
 - c. If [USSV] is 0010b:
 - i. For Base 1.x testing: set [ULSV] to 0001b.
 - ii. For Base 2.x or later testing: if [USTAGE] is 0, set [ULSV] to 0010b.
 - iii. For Base 2.x or later testing: if [USTAGE] is 1, set [ULSV] to 0001b.
 - d. If [USSV] is 0011b:
 - i. For Base 1.x testing: set [ULSV] to 0001b.
 - ii. For Base 2.x testing: if [USTAGE] is 0, set [ULSV] to 0010b.
 - iii. For Base 2.x testing: if [USTAGE] is 1, set [ULSV] to 0001b.
 - iv. For Base 3.x or later testing: if [USTAGE] is 0, set [ULSV] to 0011b.
 - v. For Base 3.x or later testing: if [USTAGE] is 1, set [ULSV] to 0010b.
 - vi. For Base 3.x or later testing: if [USTAGE] is 2, set [ULSV] to 0001b.
6. The Target Link Speed field (Link Control 2 register) is set to [ULSV] on the upstream port of the link and then the following tests are performed:
 - a. If [USSV] is 0001b: the same Target Link Speed field is read back, and if it does not return [ULSV] or 0000b, the test terminates with failure.
 - b. If [USSV] is greater than 0001b: the same Target Link Speed field is read back, and if it does not return [ULSV], the test terminates with failure.

7. Read the immediate upstream device's port (the downstream port of the link connected to the function under test) Link Capabilities register bits 3-0 value and assign it to downstream port supported speed value [DSSV] and perform the following:
 - a. If [DSSV] is 0000b, terminate the test, this is not considered a failure, but rather the test result is reported as skipped.
 - b. If [DLSV] is 0010b and [DSSV] is 0001b or less:
 - i. For Base 2.x or later testing: terminate the test, this is not considered a failure, but rather the test result is reported as skipped.
 - c. If [DLSV] is 0011b and [DSSV] is 0010b or less:
 - i. For Base 3.x or later testing: terminate the test, this is not considered a failure, but rather the test result is reported as skipped.
8. The Target Link Speed field (Link Control 2 register) is set to [DLSV] on the immediate upstream device's port (the downstream port of the link connected to the function under test). This port will be referred to as the downstream port of the link.
9. The Target Link Speed field (Link Control 2 register) is read back and if it does not return [DLSV] on the downstream port of the link, then the remainder of this test is skipped (this is not a failure, but the test result is reported as skipped).
10. Test software writes 1 to the Link Autonomous Bandwidth Status field (Link Status register) on the downstream port of the link (to clear the status).
11. Test software writes 1 to the Retrain Link field (Link Control register) on the downstream port of the link using a WORD access, while preserving all the other fields in this register.
12. Test software reads the Link Training field (Link Status register) on the downstream port of the link until it reads 0. If it does not return 0 within 1 second, then report this as a link training failure, and terminate the test with a failure.
13. Test software reads the Link Capabilities register bits 3-0 value in the function under test and assigns it to function under test supported speed value [FSSV] and performs the following:
 - a. If [FSSV] is 0000b, then the test terminates with failure.
 - b. If [FSSV] is less than [USSV], then the test terminates with failure.
 - c. If [FSSV] is less than [ULSV], then the test terminates with failure.
14. Test software reads the Current Link Speed field (Link Status register) in both the downstream port of the link and the function under test. Both values must return a value equal to one of the following two values: if [DLSV] ≤ [ULSV] then value is [DLSV]; if [DLSV] > [ULSV] then value is either [ULSV] or [DLSV], otherwise the test terminates with failure.
15. Test software reads the Link Bandwidth Notification Capability field (Link Capabilities register) on the downstream port of the link and if it returns 1, the following tests are performed:
 - a. Test software reads the Link Autonomous Bandwidth Status field (Link Status register) on the downstream port of the link. If it returns 1, terminate the test with a failure.
16. Based on the value of [DLSV] the following are performed:
 - a. If [DLSV] is 0001b:
 - i. For Base 2.x testing: set [DLSV] to 0010b.
 - ii. For Base 3.x or later testing: if [DSTAGE] is 0, increment [DSTAGE] by 1 and set [DLSV] to 0010b.
 - iii. For Base 3.x or later testing: if [DSTAGE] is 4, increment [DSTAGE] by 1 and set [DLSV] to 0011b.

- b. If [DLSV] is 0010b:
 - i. For Base 2.x testing: if [USTAGE] is 0, increment [USTAGE] by 1 and set [DLSV] to 0001b.
 - ii. For Base 2.x testing: if [USTAGE] is 1, set [USTAGE] to 0 and set [DLSV] to 0001b.
 - iii. For Base 3.x or later testing: if [DSTAGE] is 1, increment [DSTAGE] by 1 and set [DLSV] to 0011b.
 - iv. For Base 3.x or later testing: if [DSTAGE] is 3, increment [DSTAGE] by 1 and set [DLSV] to 0001b.
 - c. If [DLSV] is 0011b:
 - i. For Base 3.x or later testing: if [DSTAGE] is 2, increment [DSTAGE] by 1 and set [DLSV] to 0010b.
 - ii. For Base 3.x or later testing: if [DSTAGE] is 5 and [USTAGE] is less than 2, set [DSTAGE] to 0, increment [USTAGE] by 1, and set [DLSV] to 0001b.
 - iii. For Base 3.x or later testing: if [DSTAGE] is 5 and [USTAGE] is 2, set [DSTAGE] to 0, set [USTAGE] to 0, and set [DLSV] to 0001b.
17. Increment [ICNT] by 1.
18. Based on the value of [ICNT] the following are performed:
- a. If [ICNT] equals or is greater than 100, then terminate the test and report the test results.
 - b. If [ICNT] is less than 100, repeat steps 5-18.

The test *fails* if:

- ☐ The function under test fails to respond to any part of the standard configuration sequence.
- ☐ Link training fails to complete within 1 second of a test software initiated link retrain.
- ☐ The function under test is connected to an upstream port, but it reports a Max Link Speed/Supported Link Speeds field of 0.
- ☐ If the function under test is connected to an upstream port, is not Function 0, but Function 0 in the same multi-function device reports a Max Link Speed/Supported Link Speeds field of 0.
- ☐ If the function under test is Function 0 and it reports a Max Link Speed/Supported Link Speeds field greater than 1 and its Target Link Speed field cannot be written with a value between 1 and the value reported in the Max Link Speed/Supported Link Speeds field.
- ☐ If the function under test is not Function 0 and Function 0 in the same multi-function device reports a Max Link Speed/Supported Link Speeds field greater than 1, and Function 0's Target Link Speed field cannot be written with a value between 1 and the value reported in the Max Link Speed/Supported Link Speeds field.
- ☐ The Current Link Speed in both the upstream port and the downstream port does not match the expected link speed in any case.
- ☐ The Link Autonomous Bandwidth Status in the downstream port returns 1 in any case.

2.3.8. TD_2_8 Supported Link Width (Upstream Ports)

The test verifies that the function under test can function at the correct link speed when hardware attempts to configure the function under test with each supported link width as defined in the relevant specifications. (The test prompts the user to provide the correct physical link width for each test case. In order to successfully complete this test, the function under test must be able to pass all the standard link widths up to the maximum supported link width of the function.)

The test only runs if the downstream port reports a PCI Express Capability Version of 2h or greater.

Relevant Specifications

□ *PCI Express Base Specification*

(See Section 1.3 for additional specification revision details.)

Applicable Device/Port Types

This test is run on upstream ports of PCI Express Endpoints, Legacy Endpoints, Switch Upstream Ports, and PCI Express to PCI/PCI-X Bridges.

Starting Configuration

Function under test is completely uninitialized following a reset.

Overview of Test Steps

Test software performs the following steps:

1. Set the tested link width value [LWV] to 00 0001b (x1).
2. Set the tested link speed value [LSV] to 0001b (2.5 GT/s). Prompt the user and ask if the function under test is connected to a x1 link. (If the slot has a link that contains more than 1 physical lane, it is necessary to restrict the lane width to x1 by placing a x16 to x1 mechanical adaptor in the system slot. All lanes except the configured width for the test are not terminated during the test case.) If the user response is NO, then skip this part of the test (this is not a failure) and continue the test at step 10.
3. Configure the function under test following procedure described in Section 2.1.1.
4. Set the link speed to [LSV], following the appropriate procedures described in Sections 2.1.2.12 to 2.1.2.14.
5. Test software reads the Current Link Speed field (Link Status register) in both the function under test and the immediate upstream device's port (the downstream port of the link connected to the function under test). This port will be referred to as the downstream port of the link. The values must return [LSV], if both sides support that link speed.
6. Test software reads the Negotiated Link Width field (Link Status register) in both the downstream port of the link and the function under test. The values must return [LWV].
7. Repeat steps 3-6 for 100 iterations.
8. For Base 2.x or later testing: if both the downstream port of the link and the function under test report Max Link Speed/Supported Link Speeds (Link Capabilities register) of 0010b or greater (it is capable of 5.0 GT/s operation) then set the tested link speed value [LSV] to 0010b (5.0 GT/s) and repeat steps 3-7.

9. For Base 3.x or later testing: if both the downstream port of the link and the function under test report Max Link Speed/Supported Link Speeds (Link Capabilities register) of 0011b or greater (it is capable of 8.0 GT/s operation) then set the tested link speed value [LSV] to 0011b (8.0 GT/s) and repeat steps 3-7.
10. Set the tested link width value [LWV] to 00 0010b (x2).
 - a. If the function under test reports Maximum Link Width field (Link Capabilities register) less than [LWV], terminate the test (this is not a failure).
 - b. If the downstream port of the link reports Maximum Link Width field (Link Capabilities register) less than [LWV], terminate the test (this is not a failure).
 - c. Set the tested link speed value [LSV] to 0001b (2.5 GT/s). Prompt the user and ask if the function under test is connected to a x2 link. (If the slot has a link that contains more than 2 physical lanes, it is necessary to restrict the lane width to x2 by placing a x16 to x2 mechanical adaptor in the system slot. All lanes except the configured width for the test are not terminated during the test case.) If the user response is NO, then skip this part of the test (this is not a failure) and continue the test at step 11.
 - d. Repeat steps 3-9, with the following exception. For the first iteration only if the test fails to establish a link at the desired [LWV], then skip this part of the test (this is not a failure) and continue the test at step 11. Note: Since x2 support is optional, a port may not support this link width, but if it does support this link width, it must be able to train to it for each iteration attempt.
11. Set the tested link width value [LWV] to 00 0100b (x4).
 - a. If the function under test reports Maximum Link Width field (Link Capabilities register) less than [LWV], terminate the test (this is not a failure).
 - b. If the downstream port of the link reports Maximum Link Width field (Link Capabilities register) less than [LWV], terminate the test (this is not a failure).
 - c. Set the tested link speed value [LSV] to 0001b (2.5 GT/s). Prompt the user and ask if the function under test is connected to a x4 link. (If the slot has a link that contains more than 4 physical lanes, it is necessary to restrict the lane width to x4 by placing a x16 to x4 mechanical adaptor in the system slot. All lanes except the configured width for the test are not terminated during the test case.) If the user response is NO, then skip this part of the test (this is not a failure) and continue the test at step 12.
 - d. Repeat steps 3-9.
12. Set the tested link width value [LWV] to 00 1000b (x8).
 - a. If the function under test reports Maximum Link Width field (Link Capabilities register) less than [LWV], terminate the test (this is not a failure).
 - b. If the downstream port of the link reports Maximum Link Width field (Link Capabilities register) less than [LWV], terminate the test (this is not a failure).
 - c. Set the tested link speed value [LSV] to 0001b (2.5 GT/s). Prompt the user and ask if the function under test is connected to a x8 link. (If the slot has a link that contains more than 8 physical lanes, it is necessary to restrict the lane width to x8 by placing a x16 to x8 mechanical adaptor in the system slot. All lanes except the configured width for the test are not terminated during the test case.) If the user response is NO, then skip this part of the test (this is not a failure) and continue the test at step 13.
 - d. Repeat steps 3-9.
13. Set the tested link width value [LWV] to 00 1100b (x12).

- a. If the function under test reports Maximum Link Width field (Link Capabilities register) less than [LWV], terminate the test (this is not a failure).
 - b. If the downstream port of the link reports Maximum Link Width field (Link Capabilities register) less than [LWV], terminate the test (this is not a failure).
 - c. Set the tested link speed value [LSV] to 0001b (2.5 GT/s). Prompt the user and ask if the function under test is connected to a x12 link. (If the slot has a link that contains more than 12 physical lanes, it is necessary to restrict the lane width to x12 by placing a x16 to x12 mechanical adaptor in the system slot. All lanes except the configured width for the test are not terminated during the test case.) If the user response is NO, then skip this part of the test (this is not a failure) and continue the test at step 14.
 - d. Repeat steps 3-9, with the following exception. For the first iteration only if the test fails to establish a link at the desired [LWV], then skip this part of the test (this is not a failure) and continue the test at step 14. Note: Since x12 support is optional, a port may not support this link width, but if it does support this link width, it must be able to train to it for each iteration attempt.
14. Set the tested link width value [LWV] to 01 0000b (x16).
- a. If the function under test reports Maximum Link Width field (Link Capabilities register) less than [LWV], terminate the test (this is not a failure).
 - b. If the downstream port of the link reports Maximum Link Width field (Link Capabilities register) less than [LWV], terminate the test (this is not a failure).
 - c. Set the tested link speed value [LSV] to 0001b (2.5 GT/s). Prompt the user and ask if the function under test is connected to a x16 link. If the user response is NO, then skip this part of the test (this is not a failure).
 - d. Repeat steps 3-9.

The test *fails* if:

- ☐ The function under test fails to respond to any part of the standard configuration sequence.
- ☐ The Current Link Speed does not match the expected link speed in any case.
- ☐ The Negotiated Link Width does not match the expected link width in any case.

2.3.9. TD_2_9 Software Requested Link Equalization (Upstream Ports)

The test verifies that the function under test can function at the correct link speed when test software attempts to redo the Link Equalization procedure as defined in the relevant specifications. (The test initiates the Link Equalization procedure from the downstream port, but only if that downstream port supports 8.0 GT/s. In order to pass, the test requires that the function under test support 8.0 GT/s.)

The test will only run for Base 3.x or later testing and it will run for 100 iterations and if any single iteration fails the test will report a failure and stop.

(Note: The downstream port is required to attempt a speed change any time the Retrain Link field (Link Control register) is set to 1 and the current link speed is not equal to the Target Link Speed field. However, the actual negotiated link speed will be determined by the lowest Target Link Speed field of both the downstream port and upstream port.)

For Base 3.x testing, the test will toggle the upstream port target link speed between 8.0 GT/s, 5.0 GT/s, and 2.5 GT/s.

(Note: In using 8.0 GT/s, this test assumes that hardware initiated Link Equalization has already occurred once and completed successfully.)

The test only runs if both the function under test and the downstream port device connected to the function under test, report a PCI Express Capability Version of 2h or greater.

Relevant Specifications

□ *PCI Express Base Specification*

(See Section 1.3 for additional specification revision details.)

Applicable Device/Port Types

This test is run on upstream ports of PCI Express Endpoints, Legacy Endpoints, Switch Upstream Ports, and PCI Express to PCI/PCI-X Bridges, that report support for 8.0 GT/s.

Starting Configuration

Function under test is completely uninitialized following a reset.

Overview of Test Steps

Test software performs the following steps:

1. Configure the function under test following the procedure described in Section 2.1.1.
2. Set iteration count value [ICNT] to 0, and set upstream port iteration stage value [USTAGE] to 0.
3. For this test, the immediate upstream device's port (the downstream port of the link connected to the function under test) will be referred to as the downstream port of the link. Downstream port link speed value [DLSV] is used to control the link speed of the downstream port of the link.
4. For this test, Function 0 of the device containing the function under test will be referred to as the upstream port of the link. Upstream port link speed value [ULSV] is used to control the link speed of the upstream port of the link.
5. Test software writes a WORD to the Link Status 2 register of the upstream port of the link with the Link Equalization Request field set to 1 (to clear the status).
6. Read the Link Capabilities register bits 3-0 value from the upstream port of the link and assign it to upstream port supported speed value [USSV] and perform the following:
 - a. If [USSV] is 0000b, the test terminates with failure.
 - b. If [USSV] is 0001b, set [ULSV] to 0001b.
 - c. If [USSV] is 0010b:
 - i. If [USTAGE] is 0, set [ULSV] to 0010b.
 - ii. If [USTAGE] is 1, set [ULSV] to 0001b.
 - d. If [USSV] is 0011b:
 - i. If [USTAGE] is 0, set [ULSV] to 0011b.
 - ii. If [USTAGE] is 1, set [ULSV] to 0010b.
 - iii. If [USTAGE] is 2, set [ULSV] to 0001b.

7. The Target Link Speed field (Link Control 2 register) is set to [ULSV] on the upstream port of the link and then the following tests are performed:
 - a. If [USSV] is 0001b: the same Target Link Speed field is read back, and if it does not return [ULSV] or 0000b, the test terminates with failure.
 - b. If [USSV] is greater than 0001b: the same Target Link Speed field is read back, and if it does not return [ULSV], the test terminates with failure.
8. Read the immediate upstream device's port (the downstream port of the link connected to the function under test) Link Capabilities register bits 3-0 value and assign it to downstream port supported speed value [DSSV] and perform the following:
 - a. If [DSSV] is 0010b or less, terminate the test, this is not considered a failure, but rather the test result is reported as skipped.
 - b. If [DSSV] is 0011b or greater, then set [DLSV] to 0011b.
9. Test software writes 1 to the Perform Equalization field (Link Control 3 register) on the downstream port of the link.
10. The Target Link Speed field (Link Control 2 register) is set to [DLSV] on the immediate upstream device's port (the downstream port of the link connected to the function under test). This port will be referred to as the downstream port of the link.
11. The Target Link Speed field (Link Control 2 register) is read back and if it does not return [DLSV] on the downstream port of the link, then the remainder of this test is skipped (this is not a failure, but the test result is reported as skipped).
12. Test software writes 1 to the Link Autonomous Bandwidth Status field (Link Status register) on the downstream port of the link (to clear the status).
13. Test software writes 1 to the Retrain Link field (Link Control register) on the downstream port of the link using a WORD access, while preserving all the other fields in this register.
14. Test software reads the Link Training field (Link Status register) on the downstream port of the link until it reads 0. If it does not return 0 within 1 second, then report this as a link training failure, and terminate the test with a failure.
15. Test software reads the Link Capabilities register bits 3-0 value in the function under test and assigns it to function under test supported speed value [FSSV] and performs the following:
 - a. If [FSSV] is 0000b, then the test terminates with failure.
 - b. If [FSSV] is less than [USSV], then the test terminates with failure.
 - c. If [FSSV] is less than [ULSV], then the test terminates with failure.
16. Test software reads the Current Link Speed field (Link Status register) in both the downstream port of the link and the function under test. Both values must return a value equal to one of the following two values: if [DLSV] <= [ULSV] then value is [DLSV]; if [DLSV] > [ULSV] then value is either [ULSV] or [DLSV], otherwise the test terminates with failure. If both the returned values are correct, the common returned value is assigned to the current link speed value [CLSV].
17. Test software reads the Link Bandwidth Notification Capability field (Link Capabilities register) on the downstream port of the link and if it returns 1, the following tests are performed:
 - a. Test software reads the Link Autonomous Bandwidth Status field (Link Status register) on the downstream port of the link. If it returns 1, terminate the test with a failure.

18. Test software reads the Link Status 2 register in the upstream port of the link and performs the following tests:
 - a. If [CLSV] is 0011b:
 - i. The Equalization Complete field must be 1.
 - ii. The Equalization Phase 1 Successful field must be 1.
 - iii. If the Equalization Phase 2 Successful field returns 0, then the Equalization Phase 3 Successful field must be 0.
 - iv. If the Equalization Phase 3 Successful field returns 1, then the Equalization Phase 2 Successful field must be 1.
 - v. The Link Equalization Request field must be 0.
 - b. If [CLSV] is 0010b or less and [USSV] is 0010b or less:
 - i. The Equalization Complete field must be 0.
 - ii. The Equalization Phase 1 Successful field must be 0.
 - iii. The Equalization Phase 2 Successful field must be 0.
 - iv. The Equalization Phase 3 Successful field must be 0.
 - v. The Link Equalization Request field must be 0.
19. Based on the value of [USSV] the following are performed:
 - a. If [USSV] is 0010b:
 - i. If [USTAGE] is less than 1, increment [USTAGE] by 1.
 - ii. If [USTAGE] is 1, set [USTAGE] to 0.
 - b. If [USSV] is 0011b:
 - i. If [USTAGE] is less than 2, increment [USTAGE] by 1.
 - ii. If [USTAGE] is 2, set [USTAGE] to 0.
20. Increment [ICNT] by 1.
21. Based on the value of [ICNT] the following are performed:
 - a. If [ICNT] equals or is greater than 100, then terminate the test and report the test results.
 - b. If [ICNT] is less than 100, repeat steps 5-21.

The test *fails* if:

- ☐ The function under test fails to respond to any part of the standard configuration sequence.
- ☐ Link training fails to complete within 1 second of a test software initiated link retrain.
- ☐ The function under test is connected to an upstream port, but it reports a Max Link Speed/Supported Link Speeds field of 0.
- ☐ If the function under test is connected to an upstream port, is not Function 0, but Function 0 in the same multi-function device reports a Max Link Speed/Supported Link Speeds field of 0.
- ☐ If the function under test is Function 0 and it reports a Max Link Speed/Supported Link Speeds field greater than 1 and its Target Link Speed field cannot be written with a value between 1 and the value reported in the Max Link Speed/Supported Link Speeds field.
- ☐ If the function under test is not Function 0 and Function 0 in the same multi-function device reports a Max Link Speed/Supported Link Speeds field greater than 1, and Function 0's Target Link Speed field cannot be written with a value between 1 and the value reported in the Max Link Speed/Supported Link Speeds field.
- ☐ The Current Link Speed field in both the upstream port and the downstream port does not match the expected link speed in any case.

- ❑ The Link Autonomous Bandwidth Status field in the downstream port returns 1 in any case.
- ❑ The link speed following retrain is 8.0 GT/s and all of the following fields do not return 1 in Function 0 of the device under test: Equalization Complete; Equalization Phase 1 Successful.
- ❑ The link speed following retrain is 8.0 GT/s and Equalization Phase 3 Successful field returns 1 in Function 0 of the device under test: but the Equalization Phase 2 Successful field does not return 1 in Function 0 of the device under test. (Since the downstream port may skip both Phase 2 and Phase 3 during any equalization, the Equalization Phase 2 Successful field and the Equalization Phase 3 Successful field must either both be set or both be clear for a successful equalization.)
- ❑ The link speed following retrain is not 8.0 GT/s and all of the following fields do not return 0 in Function 0 of the device under test: Equalization Complete; Equalization Phase 1 Successful; Equalization Phase 2 Successful; Equalization Phase 3 Successful.
- ❑ The Link Equalization Request field returns 1 in Function 0 of the device under test following a link retrain.

2.4. Configuration Register Tests (Devices with Downstream Ports)

This set of tests applies only to downstream ports of Root Ports, Switches, and Bridges.

2.4.1. TD_3_1 Slot Capabilities, Slot Control, and Slot Status Registers (PCIe Cap Ver = 1)

The test verifies that if the function under test reports support for a Slot Connector, it implements the Slot Capabilities, Slot Control, and Slot Status registers as defined in the relevant specifications.

The test only runs on a downstream port or a Root Complex Event Collector that reports a PCI Express Capability Version of 1h. For a downstream port if the Slot Implemented field is 0 the Slot Capabilities, Slot Control, and Slot Status registers must be hardwired to 0 except for the Presence Detect State field in the Slot Status register which must be hardwired to 1.

Relevant Specifications

- ❑ *PCI Express Base Specification, Revision 1.1 only*

(See Section 1.3 for additional specification revision details.)

Applicable Device/Port Types

This test is run on downstream ports of Root Ports, Switch Downstream Ports, PCI/PCI-X to PCI Express Bridges, and Root Complex Event Collectors.

Starting Configuration

For a non-Root Port function under test, it is completely uninitialized following a reset, except that the bus number fields are initialized with valid numbers.

Overview of Test Steps

Test software performs the following steps:

1. Configure the function under test following the procedure described in Section 2.1.1.
2. Examine the Capability ID field for each of the function's Capabilities. Determine how many instances of the Capability ID of 10h (PCI Express Capability) are found. If more than one is found, the test terminates with a failure.
3. If the Capability ID of 10h is not found for a capability, the test terminates with a failure.
4. If a Capability ID of 10h is found for a capability, read a WORD located at offset 02h in the PCI Express Capability Structure and if the Capability Version field is equal or greater than 2h, the test terminates (this is not a failure).
5. For Base 2.x or later testing: if a Capability ID of 10h is found for a capability and if the Capability Version field is equal or less than 1h, then the test terminates and reports a failure.
6. For Base 1.x testing: if a Capability ID of 10h is found for a capability and if the Capability Version field is equal or less than 1h, the following tests are performed:
7. If the Slot Implemented field is 0, then skip to step 12.
8. Read the DWORD located at offset 14h (Slot Capabilities register) in the PCI Express Capability Structure.
9. If the Attention Indicator Present field returns 1, then the following tests are performed:
 - a. Test software writes the Attention Indicator Control field with 01b.
 - b. If the No Command Completed Support field (Slot Capabilities register) is 1, test software polls the Command Completed field (Slot Status register) until it reads back a 1. If this does not occur after 1 second of polling, then step a) is repeated. If this still does not occur after 10 repeats, this part of the test terminates with a test failure and the test continues at step 10. When the Command Completed field (Slot Status register) returns 1, test software writes the Command Completed field (Slot Status register) with 1 (to clear the status).
 - c. Test software reads the Attention Indicator Control field and checks that it returns the value previously written. If not this part of the test terminates with a test failure and the test continues at step 10.
 - d. Repeat steps a)-c) using the Attention Indicator Control field data values as follows: 10b; 11b.
 - e. Perform a default value check on the Attention Indicator Control field and confirm that it returns one of the allowed default values (01b, 10b, or 11b).
10. If the Power Indicator Present field returns 1, then the following tests are performed:
 - a. Test software writes the Power Indicator Control field with 01b.
 - b. If the No Command Completed Support field (Slot Capabilities register) is 1, test software polls the Command Completed field (Slot Status register) until it reads back a 1. If this does not occur after 1 second of polling, then step a) is repeated. If this still does not occur after 10 repeats, this part of the test terminates with a test failure and the test continues at step 11. When the Command Completed field (Slot Status register) returns 1, test software writes the Command Completed field (Slot Status register) with 1 (to clear the status).
 - c. Test software reads the Power Indicator Control field and checks that it returns the value previously written. If not this part of the test terminates with a test failure and the test continues at step 11.

- d. Repeat steps a)-c) using the Power Indicator Control field data values as follows: 10b; 11b.
 - e. Perform a default value check on the Power Indicator Control field and confirm that it returns one of the allowed default values (01b, 10b, or 11b).
11. If the Power Controller Implemented field returns 1, then the following tests are performed:
- a. Test software writes the Power Controller Control field with 0.
 - b. If the No Command Completed Support field (Slot Capabilities register) is 1, test software polls the Command Completed field (Slot Status register) until it reads back a 0. If this does not occur after 1 second of polling, then step a) is repeated. If this still does not occur after 10 repeats, this part of the test terminates with a test failure and the test continues at step 12. When the Command Completed field (Slot Status register) returns 1, test software writes the Command Completed field (Slot Status register) with 1 (to clear the status).
 - c. Test software reads the Power Controller Control field and checks that it returns the value previously written. If not this part of the test terminates with a test failure and the test continues at step 12.
 - d. Test software reads the Power Fault Detected field (Slot Status register) and checks that it returns 0. If not this part of the test terminates with a test failure and the test continues at step 12.
- (Note: Power Controller Control field data value 1 is not tested, as this would power down the hot-plug slot. There is no intention in this test to remove power from any card in any slot.)
12. The following register field characteristic checks are performed:

Slot Capabilities Register (Offset 14h) – DWORD

(for Downstream Ports with Slot Implemented as 1)

- | | |
|--|--------------|
| a. Attention Button Present
(for non-FLR testing)
(for FLR testing) | HwInit
RO |
| b. Power Controller Present
(for non-FLR testing)
(for FLR testing) | HwInit
RO |
| c. MRL Sensor Present
(for non-FLR testing)
(for FLR testing) | HwInit
RO |
| d. Attention Indicator Present
(for non-FLR testing)
(for FLR testing) | HwInit
RO |
| e. Power Indicator Present
(for non-FLR testing)
(for FLR testing) | HwInit
RO |
| f. Hot-Plug Surprise
(for non-FLR testing)
(for FLR testing) | HwInit
RO |
| g. Hot-Plug Capable
(for non-FLR testing)
(for FLR testing) | HwInit
RO |
| h. Slot Power Limit Value
(for non-FLR testing) | HwInit |

	(for FLR testing)	RO
i.	Slot Power Limit Scale	
	(for non-FLR testing)	HwInit
	(for FLR testing)	RO
j.	Electromechanical Interlock Present	
	(for non-FLR testing)	HwInit
	(for FLR testing)	RO
k.	No Command Completed Support	
	(for non-FLR testing)	HwInit
	(for FLR testing)	RO
l.	Physical Slot Number	
	(for non-FLR testing)	HwInit
	(for FLR testing)	RO

Slot Capabilities Register (Offset 14h) — DWORD

(for Root Ports with Slot Implemented as 0 or RC Event Collectors)

a.	RsvdP_31-0	RO-Zero
----	------------	---------

Slot Control Register (Offset 18h) — WORD

(for Downstream Ports with Slot Implemented as 1)

a.	Attention Button Pressed Enable	
	(if Attention Button Present is 1)	RW
	(if Attention Button Present is 0)	RW or RO-Zero
b.	Power Fault Detected Enable	RW or RO-Zero
c.	MRL Sensor Changed Enable	
	(if MRL Sensor Present is 1)	RW
	(if MRL Sensor Present is 0)	RW or RO-Zero
d.	Presence Detect Changed Enable	
	(if Hot-Plug Capable is 1)	RW
	(if Hot-Plug Capable is 0)	RW or RO-Zero
e.	Command Completed Interrupt Enable	
	(if No Command Completed Support is 0)	RW
	(if No Command Completed Support is 1)	RW or RO-Zero
f.	Hot-Plug Interrupt Enable	
	(if Hot-Plug Capable is 1)	RW
	(if Hot-Plug Capable is 0)	RW or RO-Zero
g.	Attention Indicator Control	
	(if Attention Indicator Present is 0)	RW or RO-Zero
h.	Power Indicator Control	
	(if Power Indicator Present is 0)	RW or RO-Zero

- i. Power Controller Control
(if Power Controller Present is 0) RW or RO
- j. Electromechanical Interlock Control RO-Zero
- k. Data Link Layer State Changed Enable
(if Data Link Layer Link Active Reporting Capable is 1) RW
(if Data Link Layer Link Active Reporting Capable is 0) RW or RO-Zero
- l. RsvdP_15-13 RO-Zero

Slot Control Register (Offset 18h) — WORD

(for Root Ports with Slot Implemented as 0 or RC Event Collectors)

- a. RsvdP_15-0 RO-Zero

Slot Status Register (Offset 1Ah) — WORD

(for Downstream Ports with Slot Implemented as 1)

- a. Attention Button Pressed
(if Attention Button Present is 1) RW1C
(if Attention Button Present is 0) RO-Zero
- b. Power Fault Detected
(if Power Controller Present is 1) RW1C
(if Power Controller Present is 0) RO-Zero
- c. MRL Sensor Changed
(if MRL Sensor Present is 1) RW1C
(if MRL Sensor Present is 0) RO-Zero
- d. Presence Detect Changed RW1C
- e. Command Completed
(if No Command Completed Support is 0) RW1C
(if No Command Completed Support is 1) RO-Zero
- f. MRL Sensor State RO
- g. Presence Detect State RO
- h. Electromechanical Interlock Status RO
- i. Data Link Layer State Changed RW1C
- j. RsvdZ_15-9 RO-Zero

Slot Status Register (Offset 1Ah) — WORD

(for Root Ports with Slot Implemented as 0)

- a. RsvdZ_5-0 RO-Zero
- b. Presence Detect State RO-Ones
- c. RsvdZ_15-7 RO-Zero

Slot Status Register (Offset 1Ah) — WORD

(for RC Event Collectors)

- a. RsvdP_15-0 RO-Zero

13. The following default value checks are performed:

Slot Control Register Default Value (Offset 18h) – WORD

(for Downstream Ports with Slot Implemented as 1)

- a. Attention Button Pressed Enable 0
- b. Power Fault Detected Enable 0
- c. MRL Sensor Changed Enable 0

- | | |
|---|---|
| d. Presence Detect Changed Enable | 0 |
| e. Command Completed Interrupt Enable | 0 |
| f. Hot-Plug Interrupt Enable | 0 |
| g. Data Link Layer State Changed Enable | 0 |

Slot Status Register Default Value (Offset 1Ah) – WORD

(for Downstream Ports with Slot Implemented as 1)

- | | |
|-----------------------------|---|
| a. Attention Button Pressed | 0 |
| b. Power Fault Detected | 0 |
| c. MRL Sensor Changed | 0 |
| d. Presence Detect Changed | 0 |
| e. Command Completed | 0 |

The test *fails* if:

- ☐ A PCI Express capability structure is not present.
- ☐ More than one PCI Express Capability is present.
- ☐ Any of the slot related enable fields are set by default for a non-embedded slot (for Base 1.x testing only).
- ☐ Any of the slot status change/detect fields are set by default for a slot that has just been reset (for Base 1.x testing only).
- ☐ The Attention Indicator Present field is 1, but the Power Indicator Control field cannot be written with one of the valid values, or the Command Completed field status does not return 1 (for Base 1.x testing only).
- ☐ The Power Indicator Present field is 1, but the Power Indicator Control field cannot be written with one of the valid values, or the Command Completed field status does not return 1 (for Base 1.x testing only).
- ☐ The Power Controller Implemented field is 1, but the Power Controller Control field cannot be written with the ON value (0), or the Command Completed field status is not returned, or the Power Fault Detected field status returns 1 (for Base 1.x testing only).
- ☐ Any of the register field characteristic tests fail (for Base 1.x testing only).
- ☐ Any of the default value tests fail (for Base 1.x testing only).

2.4.2. TD_3_2 Root Capabilities, Root Control, and Root Status Registers (PCIe Cap Ver = 1)

The test verifies that if the function under test is a Root Port or a Root Complex Event Collector, it implements the Root Capabilities, Root Control, and Root Status registers as defined in the relevant specifications.

The test only runs on a downstream port or a Root Complex Event Collector that reports a PCI Express Capability Version of 1h.

Relevant Specifications

- ☐ *PCI Express Base Specification, Revision 1.1 only*

(See Section 1.3 for additional specification revision details.)

Applicable Device/Port Types

This test is run on Root Ports and Root Complex Event Collectors.

Starting Configuration

For a non-Root Port function under test, it is completely uninitialized following a reset, except that the bus number fields are initialized with valid numbers.

Overview of Test Steps

Test software performs the following steps:

1. Configure the function under test following the procedure described in Section 2.1.1.
2. Examine the Capability ID field for each of the function's Capabilities. Determine how many instances of the Capability ID of 10h (PCI Express Capability) are found. If more than one is found, the test terminates with a failure.
3. If the Capability ID of 10h is not found for a capability, the test terminates with a failure.
4. If a Capability ID of 10h is found for a capability, read a WORD located at offset 02h in the PCI Express Capability Structure and if the Capability Version field is equal or greater than 2h, the test terminates (this is not a failure).
5. For Base 2.x or later testing: if a Capability ID of 10h is found for a capability and if the Capability Version field is equal or less than 1h, then the test terminates and reports a failure.
6. For Base 1.x testing: if a Capability ID of 10h is found for a capability and if the Capability Version field is equal or less than 1h, the following register field characteristic checks are performed:

Root Control Register (Offset 1Ch) — WORD

(Root Port or RC Event Collector)

- | | |
|--|---------|
| a. System Error on Correctable Error Enable | RW |
| b. System Error on Non-Fatal Error Enable | RW |
| c. System Error on Fatal Error Enable | RW |
| d. PME Interrupt Enable | RW |
| e. CRS Software Visibility Enable | |
| (for Root Ports with CRS Software Visibility as 1) | RW |
| (for Root Ports with CRS Software Visibility as 0) | RO-Zero |
| (for RC Event Collectors) | RO-Zero |
| f. RsvdP_15-5 | RO-Zero |

Root Capabilities Register (Offset 1Eh) — WORD

(Root Port or RC Event Collector)

- | | |
|----------------------------|---------|
| a. CRS Software Visibility | |
| (for Root Ports) | RO |
| (for RC Event Collectors) | RO-Zero |
| b. RsvdP_15-1 | RO-Zero |

Root Status Register (Offset 20h) — DWORD

(Root Port or RC Event Collector)

- | | |
|---------------------|---------|
| a. PME Requester ID | RO |
| b. PME Status | RW1C |
| c. PME Pending | RO |
| d. RsvdZ_31-18 | RO-Zero |

7. For Base 1.x testing: if a Capability ID of 10h is found for a capability and if the Capability Version field is equal or less than 1h, the following default value checks are performed:

Root Control Register (Offset 1Ch) — WORD

(Root Port or RC Event Collector)

- | | |
|---|---|
| a. System Error on Correctable Error Enable | 0 |
| b. System Error on Non-Fatal Error Enable | 0 |
| c. System Error on Fatal Error Enable | 0 |
| d. PME Interrupt Enable | 0 |
| e. CRS Software Visibility Enable
(for Root Ports) | 0 |

Root Status Register (Offset 20h) — DWORD

(Root Port or RC Event Collector)

- | | |
|---------------|---|
| a. PME Status | 0 |
|---------------|---|

The test *fails* if:

- ☐ A PCI Express capability structure is not present.
- ☐ More than one PCI Express Capability is present.
- ☐ Any of the register field characteristic tests fail (for Base 1.x testing only).
- ☐ Any of the default value tests fail (for Base 1.x testing only).

2.4.3. TD_3_15 ASPM Configuration Stress (Downstream Ports)

The test verifies that if the function under test implements ASPM, test software attempts to configure the function under test to a variety of different ASPM settings will not cause the device to operate abnormally, as defined in the relevant specifications.

Relevant Specifications

- ☐ *PCI Express Base Specification*

(See Section 1.3 for additional specification revision details.)

Applicable Device/Port Types

This test is run on downstream ports of Root Ports, Switch Downstream Ports, and PCI/PCI-X to PCI Express Bridges.

(Note: The immediate downstream device (the upstream port of the link connected to the function under test) must be known to operate correctly in each of its supported ASPM modes.)

Starting Configuration

For a non-Root Port function under test, it is completely uninitialized following a reset, except that the bus number fields are initialized with valid numbers.

The immediate downstream device (the upstream port of the link connected to the function under test) is completely uninitialized following a reset.

Overview of Test Steps

Test software performs the following steps:

1. Configure the function under test and the immediate downstream device (the upstream port of the link connected to the function under test following the procedure described in Section 2.1.1. If the procedure fails to complete successfully, the test terminates with failure.
2. Read the Active State Power Management (ASPM) Support field from the Link Capabilities register of the immediate downstream device (the upstream port of the link connected to the function under test (this will also be referred to as the upstream port of the link). Record this value [UASPM].
3. Read the Active State Power Management (ASPM) Support field from the Link Capabilities register of the function under test, which will be referred to as the downstream port of the link. Record this value [DASPM].
4. Set ASPM disabled (Active State Power Management (ASPM) Control = 00b in Link Control register) for both the function under test and the upstream port of the link. Repeat step 1 for 100 iterations.
5. Use the level of testing to determine the interpretation of the Active State Power Management (ASPM) Support field, and what values to write to the Active State Power Management (ASPM) Control field in the Link Control register of the upstream port of the link and the downstream port of the link. If the upstream port device is part of a multi-function device (Header Type bit 7 is 1), then the same ASPM setting value must be written to all non-VF functions in that multi-function device.
 - a. For Base 1.x testing: use Table 1 in Section 2.1.2.19.1, together with [UASPM] and [DASPM] to determine which ASPM combinations are supported, repeat step 1 for 100 iterations for each of the supported ASPM combinations in Table 1 in Section 2.1.2.19.1.
 - b. For Base 2.x or later testing: read the ASPM Optionality Compliance field from the Link Capabilities register in both the upstream port of the link and the downstream port of the link, and based on their reported values perform the following:
 - i. If the ASPM Optionality Compliance field is 0 in both the upstream port of the link and the downstream port of the link, use Table 1 in Section 2.1.2.19.1, together with [UASPM] and [DASPM] to determine which ASPM combinations are supported, repeat step 1 for 100 iterations for each of the supported ASPM combinations in Table 1 in Section 2.1.2.19.1.
 - ii. If the ASPM Optionality Compliance field is 1 in the upstream port of the link and the ASPM Optionality Compliance field is 0 in the downstream port of the link, use Table 2 in Section 2.1.2.19.2, together with [UASPM] and [DASPM] to determine which ASPM combinations are supported, repeat step 1 for 100 iterations for each of the supported ASPM combinations in Table 2 in Section 2.1.2.19.2.
 - iii. If the ASPM Optionality Compliance field is 0 in the upstream port of the link and the ASPM Optionality Compliance field is 1 in the downstream port of the link, use Table 3

- in Section 2.1.2.19.3, together with [UASPM] and [DASPM] to determine which ASPM combinations are supported, repeat step 1 for 100 iterations for each of the supported ASPM combinations in Table 3 in Section 2.1.2.19.3.
- iv. If the ASPM Optionality Compliance field is 1 in both the upstream port of the link and the downstream port of the link, use Table 4 in Section 2.1.2.19.4, together with [UASPM] and [DASPM] to determine which ASPM combinations are supported, repeat step 1 for 100 iterations for each of the supported ASPM combinations in Table 4 in Section 2.1.2.19.4.
6. For functions under test that have a link, the test is run at each of the following link speeds:
 - a. 2.5 GT/s (for devices that are capable). See Section 2.1.2.12 for details.
 - b. 5.0 GT/s (for devices that are capable). See Section 2.1.2.13 for details.
 - c. 8.0 GT/s (for devices that are capable). See Section 2.1.2.14 for details.

The test *fails* if:

- ☐ The immediate downstream device (the upstream port of the link connected to the function under test) fails to respond to any part of the standard configuration sequence.

2.4.4. TD_3_13 Secondary Bus Reset (Downstream Ports)

The test verifies that the function under test implements the Secondary Bus Reset (Bridge Control register) behavior as defined in the relevant specifications.

Relevant Specifications

- ☐ *PCI Express Base Specification*
- ☐ *PCI Express to PCI/PCI-X Bridge Specification, Revision 1.0*

(See Section 1.3 for additional specification revision details.)

Applicable Device/Port Types

This test is run on downstream ports of Root Ports, Switch Downstream Ports, and PCI/PCI-X to PCI Express Bridges.

Starting Configuration

This test is run with functions starting in the D0-Uninitialized state following the standard initialization procedures in Section 2.1.1. Testing in additional Dx states may be optionally supported. ASPM states of the upstream and downstream ports are disabled for this test.

Overview of Test Steps

Test software performs the following steps:

1. Configure the function under test and the immediate downstream device (the upstream port of the link connected to the function under test following the procedure described in Section 2.1.1.
2. Verify that a device is present on the secondary side of the Bridge, or on the downstream port of the Switch, or on the downstream port of the Root Port.
3. Modify some non-sticky field in the device on the secondary side of the Bridge, or on the downstream port of the Switch, or on the downstream port of the Root Port. (Note: Any of the following fields can be used as long as they are implemented as RW in the target: Memory Space Enable (Command register); I/O Space Enable (Command register), Bus Master Enable (Command register), Interrupt Disable (Command register). If none of these fields are

implemented as RW in the target, then other non-sticky RW fields can be used. If the test software cannot find any non-sticky RW field in the target, then this step is skipped.)

4. A reset of the secondary interface is initiated by test software through writing a 1 to the Secondary Bus Reset field in the function under test (for a Bridge, the Bridge Control register in the Bridge; for a Switch, the Bridge Control register in the Switch Downstream Port; for a Root Port, the Bridge Control register in the Root Port) using a WORD access, while preserving all the other fields in this register, and then writing a 0 using a WORD access, while preserving all the other fields in this register. (See Section 2.1.1.1.2 for details of the reset algorithm.)
5. If a non-sticky RW field was found in step 3, after the necessary delay, verify the same non-sticky field (from step 3) in the device on the secondary side of the Bridge, or on the downstream port of the Switch, or on the downstream port of the Root Port is reset to its default value. If the value has not changed to the default value, the test fails.
6. For functions under test that have a link, the test is run at each of the following link speeds:
 - a. 2.5 GT/s (for devices that are capable). See Section 2.1.2.12 for details.
 - b. 5.0 GT/s (for devices that are capable). See Section 2.1.2.13 for details.
 - c. 8.0 GT/s (for devices that are capable). See Section 2.1.2.14 for details.

The test *fails* if:

- ☐ For a Bridge, a non-sticky register field in the device on the secondary side of the Bridge fails to reset to its default settings.
- ☐ For a Switch, a non-sticky register field in the device on the downstream port of the Switch fails to reset to its default settings.
- ☐ For a Root Port, a non-sticky register field in the device on the downstream port of the Root Port fails to reset to its default settings.

2.4.5. TD_3_10 Initiated Link Speed (Downstream Ports)

The test verifies that the function under test can function at the correct link speed when test software attempts to configure the function under test with each supported link speed as defined in the relevant specifications. (The test initiates the link speed change from the downstream port which is the function under test, but only if that downstream port supports selecting that link speed using the Target Link Speed field. In order to pass, the test requires that the upstream port device connected to the function under test supports all link speeds defined in the relevant specification. In order to pass, the test requires that the function under test be able train to each link speed up to the maximum link speed supported by the function under test.)

The test will run for 100 iterations and if any single iteration fails the test will report a failure and stop. The iteration steps will be different for the different Base levels of testing. However, in all steps, any link speed change will be initiated by the downstream port. (Note: The downstream port is required to attempt a speed change any time the Retrain Link field (Link Control register) is set to 1 and the current link speed is not equal to the Target Link Speed field. However, the actual negotiated link speed will be determined by the lowest Target Link Speed field of both the downstream port and upstream port.)

- ☐ For Base 1.x testing the test will just set the link speed to 2.5 GT/s on both ports of the link, one hundred times.

- ❑ For Base 2.x testing, the test will toggle the downstream port target link speed between 2.5 GT/s and 5.0 GT/s, and after each second toggle, the test will toggle the upstream port target link speed between 5.0 GT/s and 2.5 GT/s. The sequence is: DP2.5+UP5.0=2.5; DP5.0+UP5.0=5.0; DP2.5+UP2.5=2.5; DP5.0+UP2.5=2.5 or 5.0; with the sequence repeating one hundred times.
- ❑ For Base 3.x testing, the test will step the downstream port target link speed through a six step sequence, and after each sixth step is completed, the test will toggle the upstream port target link speed between 8.0 GT/s, 5.0 GT/s, and 2.5 GT/s. The sequence is: DP2.5+UP8.0=2.5; DP5.0+UP8.0=5.0; DP8.0+UP8.0=8.0; DP5.0+UP8.0=5.0; DP2.5+UP8.0=2.5; DP8.0+UP8.0=8.0; DP2.5+UP5.0=2.5; DP5.0+UP5.0=5.0; DP8.0+UP5.0=5.0 or 8.0; DP5.0+UP5.0=5.0; DP2.5+UP5.0=2.5 or 5.0; DP8.0+UP5.0=5.0 or 8.0; DP2.5+UP2.5=2.5; DP5.0+UP2.5=2.5 or 5.0; DP8.0+UP2.5=2.5 or 8.0; DP5.0+UP2.5=2.5 or 5.0; DP2.5+UP2.5=2.5; DP8.0+UP2.5=2.5 or 8.0; with the sequence repeating for one hundred link speed changes.

(Note: For 8.0 GT/s, this test assumes that hardware initiated Link Equalization has already been performed by the function under test. The test does not attempt to perform software initiated Link Equalization.)

The test only runs if both the function under test and the upstream port device connected to the function under test, report a PCI Express Capability Version of 2h or greater.

Relevant Specifications

- ❑ *PCI Express Base Specification*

(See Section 1.3 for additional specification revision details.)

Applicable Device/Port Types

This test is run on downstream ports of Root Ports, Switch Downstream Ports, and PCI/PCI-X to PCI Express Bridges, that report a PCI Express Capability Version of 2h or greater.

Starting Configuration

For a non-Root Port function under test, it is completely uninitialized following a reset, except that the bus number fields are initialized with valid numbers.

The immediate downstream device (the upstream port of the link connected to the function under test) is completely uninitialized following a reset.

Overview of Test Steps

Test software performs the following steps:

1. Configure the function under test and the immediate downstream device (the upstream port of the link connected to the function under test) following the procedure described in Section 2.1.1.
2. Set iteration count value [ICNT] to 0, set downstream port iteration stage value [DSTAGE] to 0, and set upstream port iteration stage value [USTAGE] to 0.
3. For this test, the function under test will be referred to as the downstream port of the link. Downstream port link speed value [DLSV] is used to control the link speed of the downstream port of the link. Initially set [DLSV] to 0001b.

4. For this test, Function 0 of the immediate downstream device (the upstream port of the link connected to the function under test) will be referred to as the upstream port of the link. Upstream port link speed value [ULSV] is used to control the link speed of the upstream port of the link.
5. Read the Link Capabilities register bits 3-0 value from the upstream port of the link and assign it to upstream port supported speed value [USSV] and perform the following:
 - a. If [USSV] is 0000b, terminate the test, this is not considered a failure, but rather the test result is reported as skipped.
 - b. If [USSV] is 0001b, set [ULSV] to 0001b.
 - c. If [DLSV] is 0010b and [USSV] is 0001b or less:
 - i. For Base 2.x or later testing: terminate the test, this is not considered a failure, but rather the test result is reported as skipped.
 - d. If [USSV] is 0010b:
 - i. For Base 1.x testing: set [ULSV] to 0001b.
 - ii. For Base 2.x or later testing: if [USTAGE] is 0, set [ULSV] to 0010b.
 - iii. For Base 2.x or later testing: if [USTAGE] is 1, set [ULSV] to 0001b.
 - e. If [DLSV] is 0011b and [USSV] is 0010b or less:
 - i. For Base 3.x or later testing: terminate the test, this is not considered a failure, but rather the test result is reported as skipped.
 - f. If [USSV] is 0011b:
 - i. For Base 1.x testing: set [ULSV] to 0001b.
 - ii. For Base 2.x testing: if [USTAGE] is 0, set [ULSV] to 0010b.
 - iii. For Base 2.x testing: if [USTAGE] is 1, set [ULSV] to 0001b.
 - iv. For Base 3.x or later testing: if [USTAGE] is 0, set [ULSV] to 0011b.
 - v. For Base 3.x or later testing: if [USTAGE] is 1, set [ULSV] to 0010b.
 - vi. For Base 3.x or later testing: if [USTAGE] is 2, set [ULSV] to 0001b.
6. The Target Link Speed field (Link Control 2 register) is set to [ULSV] on the upstream port of the link and then the following tests are performed:
 - a. If [USSV] is 0001b: the same Target Link Speed field is read back, and if it does not return [ULSV] or 0000b, terminate the test. This is not considered a failure, but rather the test result is reported as skipped.
 - b. If [USSV] is greater than 0001b: the same Target Link Speed field is read back, and if it does not return [ULSV], terminate the test. This is not considered a failure, but rather the test result is reported as skipped.
7. Read the function under test's Link Capabilities register bits 3-0 value and assign it to downstream port supported speed value [DSSV] and perform the following:
 - a. If [DSSV] is 0000b, then the test terminates with failure.
 - b. If [DLSV] is 0010b or greater and [DSSV] is 0001b, then set [DSTAGE] to 0 and set [DLSV] to 0001b.
 - c. If [DLSV] is 0011b and [DSSV] is 0010b, then set [DSTAGE] to 0 and set [DLSV] to 0001b.
8. The Target Link Speed field (Link Control 2 register) is set to [DLSV] on the function under test. This port will be referred to as the downstream port of the link.
9. The Target Link Speed field (Link Control 2 register) is read back and if it does not return [DLSV] on the downstream port of the link, then the test terminates with failure.

10. Test software writes a WORD to the Link Status register with both the Link Bandwidth Management Status field and the Link Autonomous Bandwidth Status field set to 1 (to clear the status).
11. Test software writes 1 to the Retrain Link field (Link Control register) on the downstream port of the link using a WORD access, while preserving all the other fields in this register.
12. Test software reads the Link Training field (Link Status register) on the downstream port of the link until it reads 0. If it does not return 0 within 1 second, then report this as a link training failure, and terminate the test with a failure.
13. Test software reads the Current Link Speed field (Link Status register) in both the function under test and the upstream port of the link. Both values must return a value equal to one of the following two values: if [DLSV] ≤ [ULSV] then value is [DLSV]; if [DLSV] > [ULSV] then value is either [ULSV] or [DLSV], otherwise the test terminates with failure.
14. Test software reads the Link Bandwidth Notification Capability field (Link Capabilities register) on the downstream port of the link and if it returns 1, the following tests are performed:
 - a. Test software reads the Link Autonomous Bandwidth Status field (Link Status register) on the downstream port of the link. If it returns 1, terminate the test with a failure.
 - b. Test software reads the Link Bandwidth Management Status field (Link Status register) on the downstream port of the link and performs the following:
 - i. If it returns 0, terminate the test with a failure.
 - ii. If it returns 1, test software writes a WORD to the Link Status register with both the Link Bandwidth Management Status field and the Link Autonomous Bandwidth Status field set to 1 (to clear the status).
15. Based on the value of [DLSV] the following are performed:
 - a. If [DLSV] is 0001b:
 - i. For Base 2.x testing: set [DLSV] to 0010b.
 - ii. For Base 3.x or later testing: if [DSTAGE] is 0, increment [DSTAGE] by 1 and set [DLSV] to 0010b.
 - iii. For Base 3.x or later testing: if [DSTAGE] is 4, increment [DSTAGE] by 1 and set [DLSV] to 0011b.
 - b. If [DLSV] is 0010b:
 - i. For Base 2.x testing: if [USTAGE] is 0, increment [USTAGE] by 1 and set [DLSV] to 0001b.
 - ii. For Base 2.x testing: if [USTAGE] is 1, set [USTAGE] to 0 and set [DLSV] to 0001b.
 - iii. For Base 3.x or later testing: if [DSTAGE] is 1, increment [DSTAGE] by 1 and set [DLSV] to 0011b.
 - iv. For Base 3.x or later testing: if [DSTAGE] is 3, increment [DSTAGE] by 1 and set [DLSV] to 0001b.
 - c. If [DLSV] is 0011b:
 - i. For Base 3.x or later testing: if [DSTAGE] is 2, increment [DSTAGE] by 1 and set [DLSV] to 0010b.
 - ii. For Base 3.x or later testing: if [DSTAGE] is 5 and [USTAGE] is less than 2, set [DSTAGE] to 0, increment [USTAGE] by 1, and set [DLSV] to 0001b.
 - iii. For Base 3.x or later testing: if [DSTAGE] is 5 and [USTAGE] is 2, set [DSTAGE] to 0, set [USTAGE] to 0, and set [DLSV] to 0001b.
16. Increment [ICNT] by 1.

17. Based on the value of [ICNT] the following are performed:
- If [ICNT] equals or is greater than 100, then terminate the test and report the test results.
 - If [ICNT] is less than 100, repeat steps 5-17.

The test *fails* if:

- ☐ The function under test fails to respond to any part of the standard configuration sequence.
- ☐ The function under test is connected to a downstream port, but it reports a Max Link Speed/Supported Link Speeds field of 0.
- ☐ The function under test reports a Max Link Speed/Supported Link Speeds field greater than 1 and its Target Link Speed field cannot be written with a value between 1 and the value reported in the Max Link Speed/Supported Link Speeds field.
- ☐ Link training fails to complete within 1 second of a test software initiated link retrain.
- ☐ The Current Link Speed in both the upstream port and the downstream port does not match the expected link speed in any case.
- ☐ In the function under test, the Link Bandwidth Notification Capability returns 1 and the Link Autonomous Bandwidth Status returns 1 following a link retrain.
- ☐ In the function under test, the Link Bandwidth Notification Capability returns 1 and the Link Autonomous Management Status returns 0 following a link retrain.

2.4.6. TD_3_11 Supported Link Width (Downstream Ports)

The test verifies that the function under test can function at the correct link speed when hardware attempts to configure the function under test with each supported link width as defined in the relevant specifications. (The test prompts the user to provide the correct physical link width for each test case. In order to successfully complete this test, the function under test must be able to pass all the standard link widths up to the maximum supported link width of the function.)

Relevant Specifications

- ☐ *PCI Express Base Specification*

(See Section 1.3 for additional specification revision details.)

Applicable Device/Port Types

This test is run on downstream ports of Root Ports, Switch Downstream Ports, and PCI/PCI-X to PCI Express Bridges.

Starting Configuration

For a non-Root Port function under test, it is completely uninitialized following a reset, except that the bus number fields are initialized with valid numbers.

The immediate downstream device (the upstream port of the link connected to the function under test) is completely uninitialized following a reset.

Overview of Test Steps

Test software performs the following steps:

1. Set the tested link width value [LWV] to 00 0001b (x1).
2. Set the tested link speed value [LSV] to 0001b (2.5 GT/s). Prompt the user and ask if the function under test is connected to a x1 link. (If the slot has a link that contains more than 1 physical lane, it is necessary to restrict the lane width to x1 by placing a x16 to x1 mechanical adaptor in the system slot, or by using an add-in card that is only x1. All lanes except the configured width for the test are not terminated during the test case.) If the user response is NO, then skip this part of the test (this is not a failure) and continue the test at step 8.
3. Configure the function under and the immediate downstream device (the upstream port of the link connected to the function under test) following the procedure described in Section 2.1.1.
4. Set the link speed to [LSV], following the appropriate procedures described in Sections 2.1.2.12 to 2.1.2.14.
5. Test software reads the Current Link Speed field (Link Status register) in both the downstream port of the function under test and the immediate downstream device (the upstream port of the link connected to the function under test). The function under test port will be referred to as the downstream port of the link. The immediate downstream device will be referred to as the upstream port of the link. The values must return [LSV], if both sides support that link speed.
6. Test software reads the Negotiated Link Width field (Link Status register) in both the downstream port of the link and the upstream port of the link. The values must return [LWV].
7. Repeat steps 3-6 for 100 iterations.
8. For Base 2.x or later testing: if the function under test reports Max Link Speed/Supported Link Speeds (Link Capabilities register) of 0010b or greater (it is capable of 5.0 GT/s operation) then set the tested link speed value [LSV] to 0010b (5.0 GT/s) and repeat steps 3-7.
9. For Base 3.x or later testing: if the function under test reports Max Link Speed/Supported Link Speeds (Link Capabilities register) of 0011b or greater (it is capable of 8.0 GT/s operation) then set the tested link speed value [LSV] to 0011b (8.0 GT/s) and repeat steps 3-7.
10. Set the tested link width value [LWV] to 00 0010b (x2).
 - a. If the function under test reports Maximum Link Width field (Link Capabilities register) less than [LWV], terminate the test (this is not a failure).
 - b. If the upstream port of the link reports Maximum Link Width field (Link Capabilities register) less than [LWV], terminate the test (this is not a failure).
 - c. Set the tested link speed value [LSV] to 0001b (2.5 GT/s). Prompt the user and ask if the function under test is connected to a x2 link. (If the slot has a link that contains more than 2 physical lanes, it is necessary to restrict the lane width to x2 by placing a x16 to x2 mechanical adaptor in the system slot, or by using a x2 add-in card. All lanes except the configured width for the test are not terminated during the test case.) If the user response is NO, then skip this part of the test (this is not a failure) and continue the test at step 11.
 - d. Repeat steps 3-9, with the following exception. For the first iteration only if the test fails to establish a link at the desired [LWV], then skip this part of the test (this is not a failure) and continue the test at step 11. Note: Since x2 support is optional, a port may not support this link width, but if it does support this link width, it must be able to train to it for each iteration attempt.

11. Set the tested link width value [LWV] to 00 0100b (x4).
 - a. If the function under test reports Maximum Link Width field (Link Capabilities register) less than [LWV], terminate the test (this is not a failure).
 - b. If the upstream port of the link reports Maximum Link Width field (Link Capabilities register) less than [LWV], terminate the test (this is not a failure).
 - c. Set the tested link speed value [LSV] to 0001b (2.5 GT/s). Prompt the user and ask if the function under test is connected to a x4 link. (If the slot has a link that contains more than 4 physical lanes, it is necessary to restrict the lane width to x4 by placing a x16 to x4 mechanical adaptor in the system slot, or by using a x4 add-in card. All lanes except the configured width for the test are not terminated during the test case.) If the user response is NO, then skip this part of the test (this is not a failure) and continue the test at step 12.
 - d. Repeat steps 3-9.
12. Set the tested link width value [LWV] to 00 1000b (x8).
 - a. If the function under test reports Maximum Link Width field (Link Capabilities register) less than [LWV], terminate the test (this is not a failure).
 - b. If the downstream port of the link reports Maximum Link Width field (Link Capabilities register) less than [LWV], terminate the test (this is not a failure).
 - c. Set the tested link speed value [LSV] to 0001b (2.5 GT/s). Prompt the user and ask if the function under test is connected to a x8 link. (If the slot has a link that contains more than 8 physical lanes, it is necessary to restrict the lane width to x8 by placing a x16 to x8 mechanical adaptor in the system slot, or by using a x8 add-in card. All lanes except the configured width for the test are not terminated during the test case.) If the user response is NO, then skip this part of the test (this is not a failure) and continue the test at step 13.
 - d. Repeat steps 3-9.
13. Set the tested link width value [LWV] to 00 1100b (x12).
 - a. If the function under test reports Maximum Link Width field (Link Capabilities register) less than [LWV], terminate the test (this is not a failure).
 - b. If the upstream port of the link reports Maximum Link Width field (Link Capabilities register) less than [LWV], terminate the test (this is not a failure).
 - c. Set the tested link speed value [LSV] to 0001b (2.5 GT/s). Prompt the user and ask if the function under test is connected to a x12 link. (If the slot has a link that contains more than 12 physical lanes, it is necessary to restrict the lane width to x12 by placing a x16 to x12 mechanical adaptor in the system slot, or by using a x12 add-in card. All lanes except the configured width for the test are not terminated during the test case.) If the user response is NO, then skip this part of the test (this is not a failure) and continue the test at step 14.
 - d. Repeat steps 3-9, with the following exception. For the first iteration only if the test fails to establish a link at the desired [LWV], then skip this part of the test (this is not a failure) and continue the test at step 14. Note: Since x12 support is optional, a port may not support this link width, but if it does support this link width, it must be able to train to it for each iteration attempt.
14. Set the tested link width value [LWV] to 01 0000b (x16).
 - a. If the function under test reports Maximum Link Width field (Link Capabilities register) less than [LWV], terminate the test (this is not a failure).
 - b. If the downstream port of the link reports Maximum Link Width field (Link Capabilities register) less than [LWV], terminate the test (this is not a failure).

- c. Set the tested link speed value [LSV] to 0001b (2.5 GT/s). Prompt the user and ask if the function under test is connected to a x16 link. If the user response is NO, then skip this part of the test (this is not a failure).
- d. Repeat steps 3-9.

The test *fails* if:

- ☐ The slot fails to respond to any part of the standard configuration sequence.
- ☐ The Current Link Speed does not match the expected link speed in any case.
- ☐ The Negotiated Link Width does not match the expected link width in any case.

2.4.7. TD_3_12 ARI Downstream Ports Support non-Zero Device Number

The test verifies that if the function under test implements ARI Forwarding the test verifies that the downstream port with ARI Forwarding enabled allows Type 0 configuration access to non-zero Device Numbers as required in the relevant specifications. (This test requires a non-ARI device to be connected to the function under test's downstream port. It utilizes the requirement that a non-ARI device's upstream port must respond to a Configuration Read for Function Number 0, regardless of the Device Number value.)

Relevant Specifications

- ☐ *PCI Express Base Specification*
- ☐ *ECN Alternate Routing-ID Interpretation (to Base 2.0 and Base 1.1)*

(See Section 1.3 for additional specification revision details.)

Applicable Device/Port Types

This test is run on downstream ports of Root Ports and Switch Downstream ports.

Starting Configuration

For a non-Root Port function under test, it is completely uninitialized following a reset, except that the bus number fields are initialized with valid numbers.

Overview of Test Steps

Test software performs the following steps:

1. Test software checks that the function under test reports the Capability Version field (PCI Express Capabilities register) is 2h or greater. If not, the test is skipped. This is not a failure, but the test result is reported as skipped.
2. Test software checks that the function under test reports the ARI Forwarding Supported field (Device Capabilities 2 register) as 1. If not, the test is skipped. This is not a failure, but the test result is reported as skipped.
3. Test software programs the ARI Forwarding Enable field (Device Control 2 register) to 0 on the function under test's downstream port.
4. Test software sets the 16 bit Bus Number/Device Number/Function Number value [BDF] as follows: bits 15-8 = bus number of downstream port; bits 7-0 = 00h.

5. Test software issues a configuration read to the function at [BDF] for the WORD at location 00h (Vendor ID register) and it must not return FFFFh. If not the test is skipped. This is not a failure, but the test result is reported as skipped.
6. Test software checks that the function at [BDF] is a non-ARI device (by checking that it does not have an ARI Extended Capability). If it is an ARI device, the test is skipped. This is not a failure, but the test result is reported as skipped.
7. Test software programs the ARI Forwarding Enable field (Device Control 2 register) to 1 on the function under test's downstream port.
8. Test software increments [BDF] by 8h (increments Device Number). If this causes the upper 8 bits of [BDF] to change (Device Number exceeds FFh), then skip to step 10.
9. Test software issues a configuration read to the function at [BDF] for the WORD at location 00h (Vendor ID register) and it must not return FFFFh.
10. Test software programs the ARI Forwarding Enable field (Device Control 2 register) to 1 on the function under test's downstream port. (This ensures that ARI functions will remain visible to other tests.)
11. For functions under test that have a link, the test is run at each of the following link speeds:
 - a. 2.5 GT/s (for devices that are capable). See Section 2.1.2.12 for details.
 - b. 5.0 GT/s (for devices that are capable). See Section 2.1.2.13 for details.
 - c. 8.0 GT/s (for devices that are capable). See Section 2.1.2.14 for details.

The test *fails* if:

- ❑ For any non-zero Device Number, the Downstream Port returns a value of FFFFh (Unsupported Request response) when an attempt is made to read the Vendor ID of Function Number = 000b on the Bus Number of the Downstream port (reading the Vendor ID of the function connected to the Downstream Port).

2.4.8. TD_3_14 Software Initiated Link Equalization (Downstream Ports)

The test verifies that the function under test can function at the correct link speed when test software attempts to redo the Link Equalization procedure as defined in the relevant specifications. (The test initiates the Link Equalization procedure from the downstream port, but only if that downstream port supports 8.0 GT/s. In order to pass, the test requires that the upstream port device connected to the function under test support 8.0 GT/s.)

The test will only run for Base 3.x or later testing and it will run for 100 iterations and if any single iteration fails the test will report a failure and stop.

For Base 3.x testing, the test will toggle the upstream port target link speed between 8.0 GT/s, 5.0 GT/s, and 2.5 GT/s.

(Note: In using 8.0 GT/s, this test assumes that hardware initiated Link Equalization has already occurred once and completed successfully.)

The test only runs if both the function under test and the upstream port device connected to the function under test, report a PCI Express Capability Version of 2h or greater.

Relevant Specifications

□ *PCI Express Base Specification*

(See Section 1.3 for additional specification revision details.)

Applicable Device/Port Types

This test is run on downstream ports of Root Ports, Switch Downstream Ports, and PCI/PCI-X to PCI Express Bridges that report support for 8.0 GT/s.

Starting Configuration

For a non-Root Port function under test, it is completely uninitialized following a reset, except that the bus number fields are initialized with valid numbers.

The immediate downstream device (the upstream port of the link connected to the function under test) is completely uninitialized following a reset.

Overview of Test Steps

Test software performs the following steps:

1. Configure the function under test and the immediate downstream device (the upstream port of the link connected to the function under test) following the procedure described in Section 2.1.1.
2. Set iteration count value [ICNT] to 0; set upstream port iteration stage value [USTAGE] to 0.
3. For this test, the function under test will be referred to as the downstream port of the link. Downstream port link speed value [DLSV] is used to control the link speed of the downstream port of the link.
4. For this test, Function 0 of the immediate downstream device (the upstream port of the link connected to the function under test) will be referred to as the upstream port of the link. Upstream port link speed value [ULSV] is used to control the link speed of the upstream port of the link.
5. Test software writes a WORD to the Link Status 2 register of the function under test with the Link Equalization Request field set to 1 (to clear the status).
6. Read the Link Capabilities register bits 3-0 value from the upstream port of the link and assign it to upstream port supported speed value [USSV] and perform the following:
 - a. If [USSV] is 0000b, terminate the test, this is not considered a failure, but rather the test result is reported as skipped.
 - b. If [USSV] is 0001b, set [ULSV] to 0001b.
 - c. If [USSV] is 0010b:
 - i. If [USTAGE] is 0, set [ULSV] to 0010b.
 - ii. If [USTAGE] is 1, set [ULSV] to 0001b.
 - d. If [USSV] is 0011b:
 - i. If [USTAGE] is 0, set [ULSV] to 0011b.
 - ii. If [USTAGE] is 1, set [ULSV] to 0010b.
 - iii. If [USTAGE] is 2, set [ULSV] to 0001b.

7. The Target Link Speed field (Link Control 2 register) is set to [ULSV] on the upstream port of the link and then the following tests are performed:
 - a. If [USSV] is 0001b: the same Target Link Speed field is read back, and if it does not return [ULSV] or 0000b, terminate the test. This is not considered a failure, but rather the test result is reported as skipped.
 - b. If [USSV] is greater than 0001b: the same Target Link Speed field is read back, and if it does not return [ULSV], terminate the test. This is not considered a failure, but rather the test result is reported as skipped.
8. Read the function under test's Link Capabilities register bits 3-0 value and assign it to downstream port supported speed value [DSSV] and perform the following:
 - a. If [DSSV] is 0010b or less, then the test terminates with failure.
 - b. If [DSSV] is 0011b or greater, then set [DLSV] to 0011b.
9. Test software writes 1 to the Perform Equalization field (Link Control 3 register) on the downstream port of the link.
10. The Target Link Speed field (Link Control 2 register) is set to [DLSV] on the function under test. This port will be referred to as the downstream port of the link.
11. The Target Link Speed field (Link Control 2 register) is read back and if it does not return [DLSV] on the downstream port of the link, then the test terminates with failure.
12. Test software writes a WORD to the Link Status register with both the Link Bandwidth Management Status field and the Link Autonomous Bandwidth Status field set to 1 (to clear the status).
13. Test software writes 1 to the Retrain Link field (Link Control register) on the downstream port of the link using a WORD access, while preserving all the other fields in this register.
14. Test software reads the Link Training field (Link Status register) on the downstream port of the link until it reads 0. If it does not return 0 within 1 second, then report this as a link training failure, and terminate the test with a failure.
15. Test software reads the Current Link Speed field (Link Status register) in both the function under test and the upstream port of the link. Both values must return a value equal to one of the following two values: if [DLSV] ≤ [ULSV] then value is [DLSV]; if [DLSV] > [ULSV] then value is either [ULSV] or [DLSV], otherwise the test terminates with failure. If both the returned values are correct, the common returned value is assigned to the current link speed value [CLSV].
16. Test software reads the Link Bandwidth Notification Capability field (Link Capabilities register) on the downstream port of the link and if it returns 1, the following tests are performed:
 - a. Test software reads the Link Autonomous Bandwidth Status field (Link Status register) on the downstream port of the link. If it returns 1, terminate the test with a failure.
 - b. Test software reads the Link Bandwidth Management Status field (Link Status register) on the downstream port of the link and performs the following:
 - i. If it returns 0, terminate the test with a failure.
 - ii. If it returns 1, test software writes a WORD to the Link Status register with both the Link Bandwidth Management Status field and the Link Autonomous Bandwidth Status field set to 1 (to clear the status).

17. Test software reads the Link Status 2 register on the downstream port of the link and performs the following tests:
 - a. If [CLSV] is 0011b:
 - i. The Equalization Complete field must be 1.
 - ii. The Equalization Phase 1 Successful field must be 1.
 - iii. The Equalization Phase 2 Successful field must be 1.
 - iv. The Equalization Phase 3 Successful field must be 1.
 - v. The Link Equalization Request field must be 0.
 - b. If [CLSV] is 0010b or less and [USSV] is 0010b or less:
 - i. The Equalization Complete field must be 0.
 - ii. The Equalization Phase 1 Successful field must be 0.
 - iii. The Equalization Phase 2 Successful field must be 0.
 - iv. The Equalization Phase 3 Successful field must be 0.
 - v. The Link Equalization Request field must be 0.
18. Based on the value of [USSV] the following are performed:
 - a. If [USSV] is 0010b:
 - i. If [USTAGE] is less than 1, increment [USTAGE] by 1.
 - ii. If [USTAGE] is 1, set [USTAGE] to 0.
 - b. If [USSV] is 0011b:
 - i. If [USTAGE] is less than 2, increment [USTAGE] by 1.
 - ii. If [USTAGE] is 2, set [USTAGE] to 0.
19. Increment [ICNT] by 1.
20. Based on the value of [ICNT] the following are performed:
 - a. If [ICNT] equals or is greater than 100, then terminate the test and report the test results.
 - b. If [ICNT] is less than 100, repeat steps 5-20.

The test *fails* if:

- ☐ The function under test fails to respond to any part of the standard configuration sequence.
- ☐ The function under test is connected to a downstream port, but it reports a Max Link Speed/Supported Link Speeds field of 0.
- ☐ The function under test reports a Max Link Speed/Supported Link Speeds field less than 3 or its Target Link Speed field cannot be written with a value of 3.
- ☐ Link training fails to complete within 1 second of a test software initiated link retrain.
- ☐ The Current Link Speed field in both the upstream port and the downstream port does not match the expected link speed in any case.
- ☐ In the function under test, the Link Bandwidth Notification Capability field returns 1 and the Link Autonomous Bandwidth Status field returns 1 following a link retrain.
- ☐ In the function under test, the Link Bandwidth Notification Capability field returns 1 and the Link Autonomous Management Status field returns 0 following a link retrain.
- ☐ In the function under test, the link speed following retrain is 8.0 GT/s and all of the following fields do not return 1: Equalization Complete; Equalization Phase 1 Successful; Equalization Phase 2 Successful; Equalization Phase 3 Successful.
- ☐ In the function under test, the link speed following retrain is not 8.0 GT/s and all of the following fields do not return 0: Equalization Complete; Equalization Phase 1 Successful; Equalization Phase 2 Successful; Equalization Phase 3 Successful.
- ☐ In the function under test, Link Equalization Request field returns 1 following a link retrain.

A

APPENDIX A. Informational Test Details

A.1 TD_1_57x MR-IOV Extended Capability Structure

The test verifies that if the function under test reports a MR-IOV Extended Capability structure, it is implemented as defined in the relevant specifications.

Relevant Specifications

- ☐ *PCI Express Base Specification*
- ☐ *ECN PCI Express Capability Structure Expansion (to Base 1.1)*
- ☐ *ECN Function Level Reset (to Base 1.1)*
- ☐ *ECN Link Bandwidth Notification (to Base 1.1)*
- ☐ *ECN Optimized Buffer Flush/Fill (to Base 2.0 and Base 2.1)*
- ☐ *ECN ASPM Optionality (to Base 2.1)*
- ☐ *Multi-Root I/O Virtualization and Sharing Specification Revision 1.0*

(See Section 1.3 for additional specification revision details.)

Applicable Device/Port Types

This test is run on all device/port types.

This test is run on any RCRB associated with the function under test.

Starting Configuration

This test is run with functions starting in the D0-Uninitialized state following the standard initialization procedures in Section 2.1.1. Testing in additional Dx states may be optionally supported. ASPM states of the upstream and downstream ports are disabled for this test.

Overview of Test Steps

Test software performs the following steps:

1. Configure the function under test following the procedure described in Section 2.1.1.
2. Examine the Extended Capability ID field for each of the detected Extended Capabilities. Determine how many instances of the Extended Capability ID of 0011h (MR-IOV Extended Capability) are found. If more than one is found, the test terminates with a failure.
3. If the Extended Capability ID of 0011h is found in an RCRB, the test terminates with a failure.
4. If an Extended Capability ID of 0011h is found for an extended capability the following checks are performed on that extended capability structure:

5. The Capability Version field must be 1h.
6. The Next Capability Offset field must be 000h or greater than 0FFh and the lower 2 bits of this field must be 00b.
7. The device type must be PCI Express Endpoint, Legacy Endpoint, Root Complex Integrated Endpoint, Switch Upstream Port, or Switch Downstream Port.
8. If the device type is PCI Express Endpoint, Legacy Endpoint, or Root Complex Integrated Endpoint the following checks are performed:
 - a. Must not be a VF.
 - b. Test software reads a DWORD from offset 04h (MR-IOV Capabilities register) and if the Is Main BF field is 1, then the following checks are performed:
 - i. Test software reads a DWORD from offset 20h (VL Arbitration Capability and Status register) and determines the MaxVL field value which is saved as [MaxVL].
 - ii. Test software writes a 1 to the MR Enable field in MR-IOV Control (offset 08h).
 - iii. Test software writes FFh to the VL Enable field in MR-IOV Control (offset 08h).
 - iv. Test software reads MR-IOV Control register (offset 08h) and checks that the VL Enable field bits [MaxVL] to 0 each are 1 and the VL Enable field bits 7 to [MaxVL] each are 0.
 - v. Test software writes a 0 to the MR Enable field in MR-IOV Control (offset 08h).
 - vi. Test software writes FFh to the VL Enable field in MR-IOV Control (offset 08h).
 - vii. Test software reads MR-IOV Control register (offset 08h) and checks that the VL Enable field bits 0 to 7 are 01h.
 - viii. Test software reads a DWORD from offset 10h (MR-IOV VH Counts register) and determines the MaxVH field value which is saved as [MaxVH].
 - ix. Test software writes a 0 to the MR Enable field in MR-IOV Control (offset 08h).
 - x. Test software writes [MaxVH] to the NumVH field in MR-IOV VH Counts (offset 10h).
 - xi. Test software reads MR-IOV VH Counts register (offset 10h) and checks that the NumVH field is [MaxVH].
 - xii. Test software writes a 1 to the MR Enable field in MR-IOV Control (offset 08h).
 - xiii. Test software writes 00h to the NumVH field in MR-IOV VH Counts (offset 10h).
 - xiv. Test software reads MR-IOV VH Counts register (offset 10h) and checks that the NumVH field is [MaxVH].
 - xv. Test software writes a 0 to the MR Enable field in MR-IOV Control (offset 08h).
 - xvi. Test software writes 00h to the NumVH field in MR-IOV VH Counts (offset 10h).
 - xvii. Test software reads MR-IOV VH Counts register (offset 10h) and checks that the NumVH field is 00h.
 - c. Test software reads a DWORD from offset 10h (MR-IOV VH Counts register), and checks that the Function Table Entry Size field is 16 or larger.
 - d. Test software reads a DWORD from offset 14h (Function Table Offset register) and performs the following checks:
 - i. The Function Table BIR field must return one of the following values: 000b to 101b.
 - ii. The BAR pointed to by the Function Table BIR field value is read and bit 0 of that BAR must be 0 (Memory BAR).
 - iii. The Function Table Offset must be 4 or larger.

- e. Test software reads a DWORD from offset 1Ch (LVF Table Offset register) and performs the following checks:
 - i The LVF Table BIR field must return one of the following values: 000b to 101b.
 - ii The BAR pointed to by the LVF Table BIR field value is read and bit 0 of that BAR must be 0 (Memory BAR).
- f. Test software reads a DWORD from offset 04h (MR-IOV Capabilities register) and if the VL Arbitration Table Present field is 1, then the following checks are performed:
 - i Test software reads VL Arbitration Capability and Status register (offset 20h) and performs the following checks:
 - a. The VL Arbitration Capability field bits 6 to 10 are each 0.
 - b. If the VL Arbitration Capability field bit 5 is 1, then the Reference Clock field must be 00b.
 - ii Test software reads a DWORD from offset 28h (VL Arbitration Table Offset register) and performs the following checks:
 - a. The VL Arbitration Table BIR field must return one of the following values: 000b to 101b.
 - b. The BAR pointed to by the VL Arbitration Table BIR field value is read and bit 0 of that BAR must be 0 (Memory BAR).
- g. Test software reads a DWORD from offset 54h (Statistics Capability register), and if the Number of Statistics Blocks field is non-zero, the following checks are performed:
 - i The Number of Statistics Blocks field is less than or equal to 32 (including 0).
 - ii If the Number of Statistics Block field is greater than 0 and less or equal to 32, the value is stored in [NSTBK].
 - iii Test software reads a DWORD from offset 58h (Statistics Block Start/Busy register) and if [NSTBK] is less than 32, it checks that bits [NSTBK] to 31 are all 0.
 - iv If [NSTBK] is less than 32, test software writes a DWORD of FFFF FFFFh to offset 58h (Statistics Block Start/Busy register).
 - v Test software reads a DWORD from offset 58h (Statistics Block Start/Busy register) and if [NSTBK] is less than 32, it checks that bits [NSTBK] to 31 are all 0.
 - vi Test software reads a DWORD from offset 5Ch (Statistics Block Stop/Busy register) and if [NSTBK] is less than 32, it checks that bits [NSTBK] to 31 are all 0.
 - vii If [NSTBK] is less than 32, test software writes a DWORD of FFFF FFFFh to offset 5Ch (Statistics Block Stop/Busy register).
 - viii Test software reads a DWORD from offset 5Ch (Statistics Block Stop/Busy register) and if [NSTBK] is less than 32, it checks that bits [NSTBK] to 31 are all 0.

- h. Test software reads a DWORD from offset 54h (Statistics Capability register), and if the Number of Statistics Descriptors field is non-zero, the following checks are performed:
 - i. Test software reads a DWORD from offset 60h (Statistics Descriptor Table Offset register) and performs the following checks:
 - a. The Statistics Descriptor Table BIR field must return one of the following values: 000b to 101b.
 - b. The BAR pointed to by the Statistics Descriptor Table BIR field value is read and bit 0 of that BAR must be 0 (Memory BAR).
 - ii. Test software reads a DWORD from offset 54h (Statistics Capability register), and if the Number of Statistics Blocks field is non-zero, the following checks are performed:
 - iii. Test software reads a DWORD from offset 64h (Statistics Block Table Offset register) and performs the following checks:
 - a. The Statistics Block Table BIR field must return one of the following values: 000b to 101b.
 - b. The BAR pointed to by the Statistics Block Table BIR field value is read and bit 0 of that BAR must be 0 (Memory BAR).
- 9. If the device type is PCI Switch Upstream Port, or Switch Downstream Port the following checks are performed:
 - a. Test software reads a DWORD from offset 18h (MR-IOV Authorization register), and checks that the VS Authorization Bitmap Offset field is greater than 0FFh.
 - b. Test software reads a DWORD from offset 1Ch (MR-IOV Port Table Sizes register), and checks that the Port Table Entry Size field is 31 or larger.
 - c. Test software reads a DWORD from offset 20h (Port Table Offset register) and performs the following checks:
 - i. The Port Table BIR field must return one of the following values: 000b to 001b.
 - ii. The BAR pointed to by the Port Table BIR field value is read and bit 0 of that BAR must be 0 (Memory BAR).
 - iii. The Port Table Offset field must be 4 or larger.
 - d. Test software reads a DWORD from offset 24h (MR-IOV VS Table Sizes register), and checks that the VS Table Entry Size field is 3 or larger.
 - e. Test software reads a DWORD from offset 28h (VS Table Offset register) and performs the following checks:
 - i. The VS Table BIR field must return one of the following values: 000b to 001b.
 - ii. The BAR pointed to by the VS Table BIR field value is read and bit 0 of that BAR must be 0 (Memory BAR).
 - iii. The VS Table Offset field must be 4 or larger.
 - f. Test software reads a DWORD from offset 2Ch (MR-IOV VS Bridge Table Sizes register), and checks that the VS Bridge Table Entry Size field is 12 or larger.
 - g. Test software reads a DWORD from offset 30h (VS Bridge Table Offset register) and performs the following checks:
 - i. The VS Bridge Table BIR field must return one of the following values: 000b to 001b.
 - ii. The BAR pointed to by the VS Bridge Table BIR field value is read and bit 0 of that BAR must be 0 (Memory BAR).
 - h. Test software reads a DWORD from offset 34h (Statistics Capability register), and if the Number of Statistics Blocks field is non-zero, the following checks are performed:
 - i. The Number of Statistics Blocks field is less than or equal to 32 (including 0).
 - ii. If the Number of Statistics Block field is greater than 0 and less or equal to 32, the value is stored in [NSTBK].

- iii Test software reads a DWORD from offset 38h (Statistics Block Start/Busy register) and if [NSTBK] is less than 32, it checks that bits [NSTBK] to 31 are all 0.
 - iv If [NSTBK] is less than 32, test software writes a DWORD of FFFF FFFFh to offset 38h (Statistics Block Start/Busy register).
 - v Test software reads a DWORD from offset 38h (Statistics Block Start/Busy register) and if [NSTBK] is less than 32, it checks that bits [NSTBK] to 31 are all 0.
 - vi Test software reads a DWORD from offset 3Ch (Statistics Block Stop/Busy register) and if [NSTBK] is less than 32, it checks that bits [NSTBK] to 31 are all 0.
 - vii If [NSTBK] is less than 32, test software writes a DWORD of FFFF FFFFh to offset 3Ch (Statistics Block Stop/Busy register).
 - viii Test software reads a DWORD from offset 3Ch (Statistics Block Stop/Busy register) and if [NSTBK] is less than 32, it checks that bits [NSTBK] to 31 are all 0.
 - i. Test software reads a DWORD from offset 34h (Statistics Capability register), and if the Number of Statistics Descriptors field is non-zero, the following checks are performed:
 - i Test software reads a DWORD from offset 40h (Statistics Descriptor Table Offset register) and performs the following checks:
 - a. The Statistics Descriptor Table BIR field must return one of the following values: 000b to 001b.
 - b. The BAR pointed to by the Statistics Descriptor Table BIR field value is read and bit 0 of that BAR must be 0 (Memory BAR).
 - j. Test software reads a DWORD from offset 34h (Statistics Capability register), and if the Number of Statistics Blocks field is non-zero, the following checks are performed:
 - i Test software reads a DWORD from offset 44h (Statistics Block Table Offset register) and performs the following checks:
 - a. The Statistics Block Table BIR field must return one of the following values: 000b to 001b.
 - b. The BAR pointed to by the Statistics Block Table BIR field value is read and bit 0 of that BAR must be 0 (Memory BAR).
10. The following register field characteristic checks are performed:

MR-IOV Extended Capability Header (Offset 00h) — DWORD

- | | |
|---------------------------------------|----|
| a. PCI Express Extended Capability ID | RO |
| b. Capability Version | RO |
| c. Next Capability Offset | RO |

MR-IOV Capabilities Register (Offset 04h) — DWORD

(Endpoint)

- | | |
|---|--------------------------|
| a. VF Mapping Supported
(if VL Migration Supported is 1)
(if VL Migration Supported is 0) | RO-Ones
RO |
| b. VF Migration Supported | RO |
| c. VL Arbitration Table Present
(if Is Main BF is 1 and MaxVL is not zero)
(if Is Main BF is 1 and MaxVL is zero)
(if Is Main BF is 0) | RO
RO-Zero
RO-Zero |
| d. Is Main BF | RO |
| e. RsvdZ_7-4 | RO-Zero |
| f. Function Dependency Link | RO |

- | | |
|--|---------------|
| g. RsvdZ_19-16 | RO-Zero |
| h. MSI Supported | RO |
| i. MSI Vector Number
(if MSI Supported is 1)
(if MSI Supported is 0) | RO
RO-Zero |

MR-IOV Capabilities Register (Offset 04h) — DWORD

(Switch)

- | | |
|----------------------|---------|
| a. RsvdZ_20-0 | RO-Zero |
| b. MSI Vector Number | RO |

MR-IOV Control Register (Offset 08h) — DWORD

(Endpoint)

- | | |
|---|--|
| a. Function Table Interrupt Enable | RW |
| b. Statistics Interrupt Enable
(if Number of Statistic Blocks is non-zero)
(if Number of Statistics Blocks is 0) | RW
RO-Zero
RO-Zero |
| c. RsvdP_2 | RO-Zero |
| d. MR Uncorrectable Fatal TLP Error Interrupt Enable
(if Is Main BF is 1)
(if Is Main BF is 0) | RW
RO-Zero
RO-Zero |
| e. RsvdP_4 | RO-Zero |
| f. MR Uncorrectable Global Key Error Interrupt Enable
(if Is Main BF is 1)
(if Is Main BF is 0) | RW
RO-Zero
RO-Zero |
| g. RsvdP_6 | RO-Zero |
| h. MR Enable
(if Is Main BF is 1)
(if Is Main BF is 0) | RW
RO-Zero
RO-Zero |
| i. RsvdP_15-8 | RO-Zero |
| j. VL Enable
(if Is Main BF is 1: bit 0)
(if Is Main BF is 1 and MR Enable is written with 1: bits [MaxVF] to 1)
(if Is Main BF is 1 and MR Enable is written with 0: bits [MaxVF] to 1)
(if Is Main BF is 1: bits 7 to [MaxVF])
(if Is Main BF is 0)
(The MR Enable field must be written with the appropriate value before this field is tested.
Only values up to the value returned in the MaxVF register can be written.) | RO-Ones
RW
RO-Zero
RO-Zero
RO-Zero |
| k. RsvdP_31-24 | RO-Zero |

MR-IOV Control Register (Offset 08h) — DWORD

(Switch)

- | | |
|--|---------------|
| a. Port Interrupt Enable
(if VS is Authorized)
(if VS is not Authorized) | RW
RO-Zero |
| b. VS Interrupt Enable
(if VS is Authorized)
(if VS is not Authorized) | RW
RO-Zero |

- | | |
|--|--------------------------|
| c. RsvdP_2 | RO-Zero |
| d. Statistics Interrupt Enable
(if VS is Authorized and Number of Statistic Blocks is non-zero)
(if VS is Authorized and Number of Statistic Blocks is 0)
(if VS is not Authorized) | RW
RO-Zero
RO-Zero |
| e. RsvdP_15-4 | RO-Zero |
| f. MR Switch Number
(if VS is Authorized)
(if VS is not Authorized) | RW
RO-Zero |

MR-IOV Status Register (Offset 0Ch) — DWORD

(Endpoint)

- | | |
|---|-----------------|
| a. Function Table Interrupt Status | RO |
| b. Statistics Interrupt Status
(if Number of Statistic Blocks is non-zero)
(if Number of Statistic Blocks is 0) | RW1C
RO-Zero |
| c. RsvdZ_14-2 | RO-Zero |
| d. MSI Scheduled
(if MSI Supported is 1)
(if MSI Supported is 0) | RW1C
RO-Zero |
| e. VL Negotiation Pending
(if Is Main BF is 1)
(if Is Main BF is 0) | RO
RO-Zero |
| f. RsvdZ_31-24 | RO-Zero |

MR-IOV Status Register (Offset 0Ch) — DWORD

(Switch)

- | | |
|---|-----------------|
| a. Port Interrupt Status
(if VS is Authorized)
(if VS is not Authorized) | RO
RO-Zero |
| b. VS Interrupt Status
(if VS is Authorized)
(if VS is not Authorized) | RO
RO-Zero |
| c. RsvdZ_2 | RO-Zero |
| d. Statistics Interrupt Status
(if Number of Statistic Blocks is non-zero)
(if Number of Statistic Blocks is 0) | RW1C
RO-Zero |
| e. RsvdZ_14-4 | RO-Zero |
| f. MSI Scheduled | RW1C |
| g. RsvdZ_31-16 | RO-Zero |

MR-IOV VH Counts Register (Offset 10h) — DWORD

(Endpoint)

- | | |
|--|---------------|
| a. MaxVH
(if Is Main BF is 1)
(if Is Main BF is 0) | RO
RO-Zero |
| b. Function Table Entry Size | RO |
| c. NumVH | |

- | | |
|---|---------|
| (if Is Main BF is 1 and MR Enable is written with 0) | RW |
| (if Is Main BF is 1 and MR Enable is written with 1) | RO |
| (if Is Main BF is 0) | RO-Zero |
| (The MR Enable field must be written with the appropriate value before this field is tested.) | |
| d. RsvdZ_31-24 | RO-Zero |

MR-IOV This Bridge Map Register (Offset 10h) — DWORD

(Switch)

- | | |
|---------------------|---------|
| a. VS is Authorized | RO |
| b. RsvdZ_15-1 | RO-Zero |
| c. VS Bridge Number | RO |
| d. VS Number | RO |

MR-IOV Function Table Offset Register (Offset 14h) — DWORD

(Endpoint)

- | | |
|--------------------------|----|
| a. Function Table BIR | RO |
| b. Function Table Offset | RO |

MR-IOV Watchdog Timer Control Register (Offset 14h) — DWORD

(Switch)

- | | |
|----------------------------|---------|
| a. Rearm Watchdog | RO |
| b. RsvdP_15-1 | RO-Zero |
| c. Watchdog Timer Interval | |
| (if VS is Authorized) | RW |
| (if VS is not Authorized) | RO-Zero |
| d. RsvdP_31-24 | RO-Zero |

(If the Watchdog Timer Interval field is written with a non-zero value, then the Rearm Watchdog field must be written with a 1 every second until the Watchdog Timer Interval field is written to 0. Failure to do this may cause the Switch to get reset to its default state if the Watchdog timer expires.)

MR-IOV VF MVF Region Register (Offset 18h) — DWORD

(Endpoint)

- | | |
|--------------------------------|---------|
| a. Max LVF | RO |
| b. Max MVF | |
| (if VF Mapping Supported is 1) | RO |
| (if VF Mapping Supported is 0) | RO-Zero |

MR-IOV Authorization Register (Offset 18h) — DWORD

(Switch)

- | | |
|-----------------------------------|---------|
| a. Management VS | |
| (if VS is Authorized) | RW |
| (if VS is not Authorized) | RO-Zero |
| b. RsvdZ_19-8 | RO-Zero |
| c. VS Authorization Bitmap Offset | RO |

MR-IOV LVF Table Offset Register (Offset 1Ch) — DWORD

(Endpoint)

- | | |
|------------------|----|
| a. LVF Table BIR | RO |
|------------------|----|

- | | |
|---------------------|----|
| b. LVF Table Offset | RO |
|---------------------|----|

MR-IOV Port Table Sizes Register (Offset 1Ch) — DWORD

(Switch)

- | | |
|---------------------------|---------|
| a. Num Port Table Entries | RO |
| b. Port Table Entry Size | RO |
| c. RsvdZ_31-16 | RO-Zero |

MR-IOV VL Arbitration Capability and Status Register (Offset 20h) — DWORD

(Endpoint)

- | | |
|--|---------|
| a. VL Arbitration Capability
(if VL Arbitration Table Present is 1) | RO |
| (if VL Arbitration Table Present is 0) | RO-Zero |
| b. VL Arbitration Status
(if VL Arbitration Table Present is 1) | RO |
| (if VL Arbitration Table Present is 0) | RO-Zero |
| c. RsvdZ_13 | RO-Zero |
| d. Reference Clock
(if VL Arbitration Table Present is 1) | RO |
| (if VL Arbitration Table Present is 0) | RO-Zero |
| e. MaxVL
(if Is Main BF is 1) | RO |
| (if Is Main BF is 0) | RO-Zero |
| f. RsvdZ_23-19 | RO-Zero |
| g. Maximum Time Slots
(if VL Arbitration Table Present is 1) | RO |
| (if VL Arbitration Table Present is 0) | RO-Zero |
| h. RsvdZ_31 | RO-Zero |

MR-IOV Port Table Offset Register (Offset 20h) — DWORD

(Switch)

- | | |
|----------------------|----|
| a. Port Table BIR | RO |
| b. Port Table Offset | RO |

MR-IOV VL Arbitration Control Register (Offset 24h) — DWORD

(Endpoint)

- | | |
|--|---------|
| a. Load VL Arbitration Table | RO-Zero |
| b. RsvdP_7-1 | RO-Zero |
| c. VL Arbitration Select
(if VL Arbitration Table Present is 1)
(only those values in VL Arbitration Capability are written)
(if VL Arbitration Table Present is 0) | RW |
| d. RsvdP_15-12 | RO-Zero |
| e. VL Strict Priority Arbitration
(if VL Arbitration Table Present is 1)
(if VL Arbitration Table Present is 0) | RW |
| f. RsvdP_31-24 | RO-Zero |

MR-IOV VS Table Sizes Register (Offset 24h) — DWORD

(Switch)

- | | |
|-------------------------|---------|
| a. Num VS Table Entries | RO |
| b. VS Table Entry Size | RO |
| c. RsvdZ_31-16 | RO-Zero |

MR-IOV VL Arbitration Table Offset Register (Offset 28h) — DWORD

(Endpoint)

- | | |
|--|---------|
| a. VL Arbitration Table BIR | |
| (if VL Arbitration Table Present is 1) | RO |
| (if VL Arbitration Table Present is 0) | RO-Zero |
| b. VL Arbitration Table Offset | |
| (if VL Arbitration Table Present is 1) | RO |
| (if VL Arbitration Table Present is 0) | RO-Zero |

MR-IOV VS Table Offset Register (Offset 28h) — DWORD

(Switch)

- | | |
|--------------------|----|
| a. VS Table BIR | RO |
| b. VS Table Offset | RO |

MR-IOV MR Error Status Register (Offset 2Ch) — WORD

(Endpoint)

- | | |
|---|---------|
| a. MR First Error Pointer | |
| (if Is Main BF is 1) | ROS |
| (if Is Main BF is 0) | RO-Zero |
| b. MR Uncorrectable Fatal TLP Error Status | |
| (if Is Main BF is 1) | RW1CS |
| (if Is Main BF is 0) | RO-Zero |
| c. RsvdZ_5 | RO-Zero |
| d. MR Uncorrectable Global Key Error Status | |
| (if Is Main BF is 1) | RW1CS |
| (if Is Main BF is 0) | RO-Zero |
| e. RsvdZ_14-7 | RO-Zero |
| f. MR Multiple Uncorrectable Error | |
| (if Is Main BF is 1) | RW1CS |
| (if Is Main BF is 0) | RO-Zero |

MR-IOV MR Error Control Register (Offset 2Eh) — WORD

(Endpoint)

- | | |
|---|---------|
| a. RsvdP_3-0 | RO-Zero |
| b. MR Uncorrectable Fatal TLP Error Mask | |
| (if Is Main BF is 1) | RWS |
| (if Is Main BF is 0) | RO-Zero |
| c. RsvdP_5 | RO-Zero |
| d. MR Uncorrectable Global Key Error Mask | |
| (if Is Main BF is 1) | RWS |
| (if Is Main BF is 0) | RO-Zero |
| e. RsvdP_15-7 | RO-Zero |

MR-IOV VS Bridge Table Sizes Register (Offset 2Ch) — DWORD

(Switch)

- | | |
|--------------------------------|---------|
| a. Num VS Bridge Table Entries | RO |
| b. VS Bridge Table Entry Size | RO |
| c. RsvdZ_31-16 | RO-Zero |

MR-IOV Header Log Register (Offset 30h) — 9 DWORDS

(Endpoint)

- | | |
|---|----------------|
| a. Header Log register (1st DW)
(if Is Main BF is 1)
(if Is Main BF is 0) | ROS
RO-Zero |
| b. Header Log register (2nd DW)
(if Is Main BF is 1)
(if Is Main BF is 0) | ROS
RO-Zero |
| c. Header Log register (3rd DW)
(if Is Main BF is 1)
(if Is Main BF is 0) | ROS
RO-Zero |
| d. Header Log register (4th DW)
(if Is Main BF is 1)
(if Is Main BF is 0) | ROS
RO-Zero |
| e. Header Log register (5th DW)
(if Is Main BF is 1)
(if Is Main BF is 0) | ROS
RO-Zero |
| f. Header Log register (6th DW)
(if Is Main BF is 1)
(if Is Main BF is 0) | ROS
RO-Zero |
| g. Header Log register (7th DW)
(if Is Main BF is 1)
(if Is Main BF is 0) | ROS
RO-Zero |
| h. Header Log register (8th DW)
(if Is Main BF is 1)
(if Is Main BF is 0) | ROS
RO-Zero |
| i. Header Log register (9th DW)
(if Is Main BF is 1)
(if Is Main BF is 0) | ROS
RO-Zero |

MR-IOV VS Bridge Table Offset Register (Offset 30h) — DWORD

(Switch)

- | | |
|---------------------------|----|
| a. VS Bridge Table BIR | RO |
| b. VS Bridge Table Offset | RO |

MR-IOV Statistics Capability Register (Offset 34h) — DWORD

(Switch)

- | | |
|---|---------------|
| a. Number of Statistics Descriptors
(if VS is Authorized)
(if VS is not Authorized) | RO
RO-Zero |
| b. Number of Statistics Blocks
(if VS is Authorized)
(if VS is not Authorized) | RO
RO-Zero |
| c. RsvdZ_31-16 | RO-Zero |

MR-IOV Statistics Block Start/Busy Register (Offset 38h) — DWORD

(Switch)

(if VS is not Authorized then all bits)

RO-Zero

(if Number of Statistics Blocks is 00h then all bits)

RO-Zero

(if Number of Statistics Blocks is non-zero and less than 32:

bits 31-[Number of Statistics Blocks])

RO-Zero

(For the bits that are implemented the value written cannot be read back, so only the unimplemented bits in this register can be checked.)

MR-IOV Statistics Block Stop/Busy Register (Offset 3Ch) — DWORD

(Switch)

(if VS is not Authorized then all bits)

RO-Zero

(if Number of Statistics Blocks is 00h then all bits)

RO-Zero

(if Number of Statistics Blocks is non-zero and less than 32:

bits 31-[Number of Statistics Blocks])

RO-Zero

(For the bits that are implemented the value written cannot be read back, so only the unimplemented bits in this register can be checked.)

MR-IOV Statistics Descriptor Table Offset Register (Offset 40h) — DWORD

(Switch)

a. Statistics Descriptor Table BIR

(if Number of Statistics Descriptors is non-zero)

RO

(if Number of Statistics Descriptors is 00h)

RO-Zero

b. Statistics Descriptor Table Offset

(if Number of Statistics Descriptors is non-zero)

RO

(if Number of Statistics Descriptors is 00h)

RO-Zero

MR-IOV Statistics Block Table Offset Register (Offset 44h) — DWORD

(Switch)

a. Statistics Block Table BIR

(if Number of Statistics Blocks is non-zero)

RO

(if Number of Statistics Blocks is 00h)

RO-Zero

b. Statistics Block Table Offset

(if Number of Statistics Blocks is non-zero)

RO

(if Number of Statistics Blocks is 00h)

RO-Zero

MR-IOV Statistics Capability Register (Offset 54h) — DWORD

(Endpoint)

a. Number of Statistics Descriptors

RO

b. Number of Statistics Blocks

RO

c. RsvdZ_31-16

RO-Zero

MR-IOV Statistics Block Start/Busy Register (Offset 58h) — DWORD

(Endpoint)

(if Number of Statistics Blocks is 00h then all bits)

RO-Zero

(if Number of Statistics Blocks is non-zero and less than 32:

bits 31-[Number of Statistics Blocks])

RO-Zero

(For the bits that are implemented the value written cannot be read back, so only the unimplemented bits in this register can be checked.)

MR-IOV Statistics Block Stop/Busy Register (Offset 5Ch) — DWORD

(Endpoint)

(if Number of Statistics Blocks is 00h then all bits)

RO-Zero

(if Number of Statistics Blocks is non-zero and less than 32:

bits 31-[Number of Statistics Blocks])

RO-Zero

(For the bits that are implemented the value written cannot be read back, so only the unimplemented bits in this register can be checked.)

MR-IOV Statistics Descriptor Table Offset Register (Offset 60h) — DWORD

(Endpoint)

- a. Statistics Descriptor Table BIR

(if Number of Statistics Descriptors is non-zero)

RO

(if Number of Statistics Descriptors is 00h)

RO-Zero

- b. Statistics Descriptor Table Offset

(if Number of Statistics Descriptors is non-zero)

RO

(if Number of Statistics Descriptors is 00h)

RO-Zero

MR-IOV Statistics Block Table Offset Register (Offset 64h) — DWORD

(Endpoint)

- a. Statistics Block Table BIR

(if Number of Statistics Blocks is non-zero)

RO

(if Number of Statistics Blocks is 00h)

RO-Zero

- b. Statistics Block Table Offset

(if Number of Statistics Blocks is non-zero)

RO

(if Number of Statistics Blocks is 00h)

RO-Zero

VS Authorization Bitmap ([offset]) – [r] DWORDS

(Switch)

- a. ([Num VS Table Entries]-1 to 0)

(for bit [n] if VS Table Entry [n] VS Present is 1 and VS is Authorized)

RW

Note: All values of [n] from 0 to [Num VS Table Entries]-1 are tested.

(for bit [n] if VS Table Entry [n] VS Present is 0)

RO-Zero

Note: All values of [n] from 0 to [Num VS Table Entries]-1 are tested.

(if VS is not Authorized)

RO-Zero

- b. (bits ([r] * 32) – 1 to [Num VS Table Entries])

RO-Zero

(This group of registers only exists if the Num VS Table Entries field is greater than 0, and therefore this group of registers is not tested if this condition is not true.)

Note: The value of [offset] is the value returned by the VS Authorization Bitmap Offset field. The value of [r] is ([Num VS Table Entries] divided by 32) rounded up to the next integer number.

LVF Table LVF Table Entry (n) ([table] + ([n] * 4)) — DWORD

(Endpoint)

- | | | |
|----|---|---------|
| a. | MVF # for this LVF | |
| | (bits [high bit] to 0) | RW |
| | (bits 15 to [high bit] + 1) | RO-Zero |
| | (The value of [high bit] is the smallest value that satisfies the requirement that $2^{**}[\text{high bit}]$ is able to denote the range 0 to [Max MVF].) | |
| b. | VF State | |
| | (if VF Migration Supported is 1) | RW |
| | (if VF Migration Supported is 0) | RO-Zero |
| c. | RsvdP_31-18 | RO-Zero |

(This group of registers only exists if the VF Mapping Supported field is a 1, and therefore this group of registers is not tested if this condition is not true. The MR Enable field must be written with 0, then the NumVH field must be written with the value returned by the MaxVH field, and then the MR Enable field must be written with 1 before this register is tested.)

Note: Testing this requires programming the designated Memory BAR and enabling Memory space. The value of [table] is the BAR contents (pointed to by the LVF Table BIR field value) plus the offset (value in the LVF Table Offset field multiplied by 8). All values of [n] between 0 and the value in the Max LVF field inclusive must be tested.

Function Interrupt Status Bitmap ([table] - 8) — 8 DWORDS

(Endpoint)

- | | |
|---|---------|
| (if MSI Supported is 1 then for all bits) | RO |
| (if MSI Supported is 0 then for all bits) | RO-Zero |

(The MR Enable field must be written with 0, then the NumVH field must be written with the value returned by the MaxVH field, and then the MR Enable field must be written with 1 before this register is tested.)

Note: Testing this requires programming the designated Memory BAR and enabling Memory space. The value of [table] is the BAR contents (pointed to by the Function Table BIR field value) plus the offset (value in the Function Table Offset field multiplied by 8).

Function Table Entry (n) Function Capabilities 1 ([table] + ([n] * [size])) + 00h — DWORD

(Endpoint)

- | | | |
|----|-------------------------------------|---------|
| a. | Num VC Resources Hardware Present | |
| | (if VC Capability Supported is 1) | RO |
| | (if VC Capability Supported is 0) | RO-Zero |
| b. | RsvdZ_3 | RO-Zero |
| c. | VC Capability Supported | RO |
| d. | RsvdZ_7-5 | RO-Zero |
| e. | Num MFVC Resources Hardware Present | |
| | (if MFVC Capability Supported is 1) | RO |
| | (if MFVC Capability Supported is 0) | RO-Zero |
| f. | RsvdZ_11 | RO-Zero |

g. MFVC Capability Supported	RO
h. RsvdZ_15-13	RO-Zero
i. Function Present	
(if (n) is 0)	RO
(if (n) is non-zero)	RO-Zero
j. RsvdZ_23-17	RO-Zero
k. Function Offset	RO

(The MR Enable field must be written with 0, then the NumVH field must be written with the value returned by the MaxVH field, and then the MR Enable field must be written with 1 before this register is tested.)

Note: Testing this requires programming the designated Memory BAR and enabling Memory space. The value of [table] is the BAR contents (pointed to by the Function Table BIR field value) plus the offset (value in the Function Table Offset field multiplied by 8). The value of [size] is the value in the Function Table Entry Size field multiplied by 4. All values of [n] between 0 and the value in the MaxVH field inclusive must be tested.

Function Table Entry (n) Function Capabilities 2 ([table] + ([n] * [size])) + 04h — DWORD

(Endpoint)

a. First VF Offset	
(for PFs)	RO
(for non-PFs)	RO-Zero
b. VF Stride	
(for PFs)	RO
(for non-PFs)	RO-Zero

(The MR Enable field must be written with 0, then the NumVH field must be written with the value returned by the MaxVH field, and then the MR Enable field must be written with 1 before this register is tested.)

Note: Testing this requires programming the designated Memory BAR and enabling Memory space. The value of [table] is the BAR contents (pointed to by the Function Table BIR field value) plus the offset (value in the Function Table Offset field multiplied by 8). The value of [size] is the value in the Function Table Entry Size field multiplied by 4. All values of [n] between 0 and the value in the MaxVH field inclusive must be tested.

Function Table Entry (n) Function Control 1 ([table] + ([n] * [size])) + 08h — DWORD

(Endpoint)

a. VC Extended VC Count	
(if VC Capability Supported is 1)	RW
(if VC Capability Supported is 0)	RO-Zero
b. RsvdP_3	RO-Zero
c. VC Low Priority Extended VC Count	
(if VC Capability Supported is 1)	RW
(if VC Capability Supported is 0)	RO-Zero
d. RsvdP_7	RO-Zero
e. MFVC Extended VC Count	
(if MFVC Capability Supported is 1)	RW
(if MFVC Capability Supported is 0)	RO-Zero
f. RsvdP_11	RO-Zero

- | | | |
|----|---|--------------------------|
| g. | MFVC Low Priority Extended VC Count
(if MFVC Capability Supported is 1)
(if MFVC Capability Supported is 0) | RW
RO-Zero
RO-Zero |
| h. | RsvdP_15 | RO-Zero |
| i. | Global Key
(if Is Main BF is 1)
(if Is Main BF is 0) | RW
RO-Zero
RO-Zero |
| j. | RsvdP_30-28 | RO-Zero |
| k. | Global Key Check Enable
(if Is Main BF is 1)
(if Is Main BF is 0) | RW
RO-Zero
RO-Zero |

(The MR Enable field must be written with 0, then the NumVH field must be written with the value returned by the MaxVH field, and then the MR Enable field must be written with 1 before this register is tested.)

Note: Testing this requires programming the designated Memory BAR and enabling Memory space. The value of [table] is the BAR contents (pointed to by the Function Table BIR field value) plus the offset (value in the Function Table Offset field multiplied by 8). The value of [size] is the value in the Function Table Entry Size field multiplied by 4. All values of [n] between 0 and the value in the MaxVH field inclusive must be tested.

Function Table Entry (n) Function Control 2 ([table] + ([n] * [size])) + 0Ch — DWORD
(Endpoint)

- | | | |
|----|---|--------------------------|
| a. | VC Config Changed Interrupt Enable
(if VC Capability Supported is 1)
(if VC Capability Supported is 0) | RW
RO-Zero |
| b. | MFVC Config Changed Interrupt Enable
(if MFVC Capability Supported is 1)
(if MFVC Capability Supported is 0) | RW
RO-Zero |
| c. | PF Reset Initiated Interrupt Enable
(if VF Migration Supported is 1)
(if VF Migration Supported is 0) | RW
RO-Zero |
| d. | VF Migration Status Interrupt Enable
(if VF Migration Supported is 1)
(if VF Migration Supported is 0) | RW
RO-Zero |
| e. | VF Enable Interrupt Enable
(if VF Mapping Supported is 1)
(if VF Mapping Supported is 0) | RW
RO-Zero
RO-Zero |
| f. | RsvdP_8-5 | RO-Zero |
| g. | VF Migration Capable
(if VF Migration Supported is 1)
(if VF Migration Supported is 0) | RW
RO-Zero
RO-Zero |
| h. | RsvdP_15-10 | RO-Zero |
| i. | InitialVFs
(for PFs with VF Mapping Supported as 1)
(for PFs with VF Mapping Supported as 0)
(for non-PFs) | RW
RO
RO-Zero |

(The MR Enable field must be written with 0, then the NumVH field must be written with the value returned by the MaxVH field, and then the MR Enable field must be written with 1 before this register is tested.)

Note: Testing this requires programming the designated Memory BAR and enabling Memory space. The value of [table] is the BAR contents (pointed to by the Function Table BIR field value) plus the offset (value in the Function Table Offset field multiplied by 8). The value of [size] is the value in the Function Table Entry Size field multiplied by 4. All values of [n] between 0 and the value in the MaxVH field inclusive must be tested.

Function Table Entry (n) Function Control 3 ([table] + ([n] * [size])) + 10h — DWORD
(Endpoint)

- | | |
|----------------------------------|---------|
| a. Base LVF | |
| (if VF Mapping Supported is 1) | RW |
| (if VF Mapping Supported is 0) | RO-Zero |
| b. TotalVFs | |
| (if VF Migration Supported is 1) | RW |
| (if VF Migration Supported is 0) | RO-Zero |

(The MR Enable field must be written with 0, then the NumVH field must be written with the value returned by the MaxVH field, and then the MR Enable field must be written with 1 before this register is tested.)

Note: Testing this requires programming the designated Memory BAR and enabling Memory space. The value of [table] is the BAR contents (pointed to by the Function Table BIR field value) plus the offset (value in the Function Table Offset field multiplied by 8). The value of [size] is the value in the Function Table Entry Size field multiplied by 4. All values of [n] between 0 and the value in the MaxVH field inclusive must be tested.

Function Table Entry (n) Function Status ([table] + ([n] * [size])) + 14h — DWORD
(Endpoint)

- | | |
|-------------------------------------|---------|
| a. VC Config Changed | |
| (if VC Capability Supported is 1) | RW1C |
| (if VC Capability Supported is 0) | RO-Zero |
| b. MFVC Config Changed | |
| (if MFVC Capability Supported is 1) | RW1C |
| (if MFVC Capability Supported is 0) | RO-Zero |
| c. PF Reset Initiated | |
| (if VF Migration Supported is 1) | RW1C |
| (if VF Migration Supported is 0) | RO-Zero |
| d. VF Migration Status | |
| (if VF Migration Supported is 1) | RW1C |
| (if VF Migration Supported is 0) | RO-Zero |
| e. VF Enable Changed | |
| (if VF Mapping Supported is 1) | RW1C |
| (if VF Mapping Supported is 0) | RO-Zero |
| f. VF Initialization Pending | |
| (if VF Migration Supported is 1) | RW1C |
| (if VF Migration Supported is 0) | RO-Zero |
| g. RsvdZ_7-6 | RO-Zero |
| h. VF Enabled | RO |

- | | |
|----------------------------------|---------|
| i. VF Migration Enabled | |
| (if VF Migration Supported is 1) | RO |
| (if VF Migration Supported is 0) | RO-Zero |
| j. RsvdZ_15-10 | RO-Zero |
| k. NumVFs | RO |

(The MR Enable field must be written with 0, then the NumVH field must be written with the value returned by the MaxVH field, and then the MR Enable field must be written with 1 before this register is tested.)

Note: Testing this requires programming the designated Memory BAR and enabling Memory space. The value of [table] is the BAR contents (pointed to by the Function Table BIR field value) plus the offset (value in the Function Table Offset field multiplied by 8). The value of [size] is the value in the Function Table Entry Size field multiplied by 4. All values of [n] between 0 and the value in the MaxVH field inclusive must be tested.

Function Table Entry (n) VC to VL Map 1 ([table] + ([n] * [size])) + 18h — DWORD

(Endpoint)

- | | |
|---|---------|
| a. VC0 VL Map | |
| (if Is Main BF is 1 and MaxVL is non-zero and VC Capability Supported is 1) | |
| after first writing VC0 VL Map Enable with 0 | RW |
| after first writing VC0 VL Map Enable with 1 | RO |
| (if Is Main BF is 1 and MaxVL is 000b) | RO-Zero |
| (if Is Main BF is 1 and VC Capability Supported is 0) | RO-Zero |
| (if Is Main BF is 0) | RO-Zero |
| b. RsvdP_3 | RO-Zero |
| c. VC0 VL Map Enable | |
| (if Is Main BF is 1 and MaxVL is non-zero and VC Capability Supported is 1) | RW |
| (if Is Main BF is 1 and MaxVL is 000b) | RO |
| (if Is Main BF is 1 and VC Capability Supported is 0) | RO |
| (if Is Main BF is 0) | RO-Zero |
| d. RsvdP_7-5 | RO-Zero |
| e. VC1 VL Map | |
| (if Is Main BF is 1 and MaxVL is non-zero and VC Capability Supported is 1 and Num VC Resources Hardware Present is 1 or greater) | |
| after first writing VC1 VL Map Enable with 0 | RW |
| after first writing VC1 VL Map Enable with 1 | RO |
| (if Is Main BF is 1 and MaxVL is 000b) | RO-Zero |
| (if Is Main BF is 1 and VC Capability Supported is 0) | RO-Zero |
| (if Is Main BF is 1 and Num VC Resources Hardware Present is less than 1) | RO-Zero |
| (if Is Main BF is 0) | RO-Zero |
| f. RsvdP_11 | RO-Zero |
| g. VC1 VL Map Enable | |
| (if Is Main BF is 1 and MaxVL is non-zero and VC Capability Supported is 1 and Num VC Resources Hardware Present is 1 or greater) | RW |
| (if Is Main BF is 1 and MaxVL is 000b) | RO-Zero |
| (if Is Main BF is 1 and VC Capability Supported is 0) | RO-Zero |

	(if Is Main BF is 1 and Num VC Resources Hardware Present is less than 1)	RO-Zero
	(if Is Main BF is 0)	RO-Zero
h.	RsvdP_15-13	RO-Zero
i.	VC2 VL Map	
	(if Is Main BF is 1 and MaxVL is non-zero and VC Capability Supported is 1 and Num VC Resources Hardware Present is 2 or greater)	
	after first writing VC2 VL Map Enable with 0	RW
	after first writing VC2 VL Map Enable with 1	RO
	(if Is Main BF is 1 and MaxVL is 000b)	RO-Zero
	(if Is Main BF is 1 and VC Capability Supported is 0)	RO-Zero
	(if Is Main BF is 1 and Num VC Resources Hardware Present is less than 2)	RO-Zero
	(if Is Main BF is 0)	RO-Zero
j.	RsvdP_19	RO-Zero
k.	VC2 VL Map Enable	
	(if Is Main BF is 1 and MaxVL is non-zero and VC Capability Supported is 1 and Num VC Resources Hardware Present is 2 or greater)	RW
	(if Is Main BF is 1 and MaxVL is 000b)	RO-Zero
	(if Is Main BF is 1 and VC Capability Supported is 0)	RO-Zero
	(if Is Main BF is 1 and Num VC Resources Hardware Present is less than 2)	RO-Zero
	(if Is Main BF is 0)	RO-Zero
l.	RsvdP_23-21	RO-Zero
m.	VC3 VL Map	
	(if Is Main BF is 1 and MaxVL is non-zero and VC Capability Supported is 1 and Num VC Resources Hardware Present is 3 or greater)	
	after first writing VC3 VL Map Enable with 0	RW
	after first writing VC3 VL Map Enable with 1	RO
	(if Is Main BF is 1 and MaxVL is 000b)	RO-Zero
	(if Is Main BF is 1 and VC Capability Supported is 0)	RO-Zero
	(if Is Main BF is 1 and Num VC Resources Hardware Present is less than 3)	RO-Zero
	(if Is Main BF is 0)	RO-Zero
n.	RsvdP_27	RO-Zero
o.	VC3 VL Map Enable	
	(if Is Main BF is 1 and MaxVL is non-zero and VC Capability Supported is 1 and Num VC Resources Hardware Present is 3 or greater)	RW
	(if Is Main BF is 1 and MaxVL is 000b)	RO-Zero
	(if Is Main BF is 1 and VC Capability Supported is 0)	RO-Zero
	(if Is Main BF is 1 and Num VC Resources Hardware Present is less than 3)	RO-Zero
	(if Is Main BF is 0)	RO-Zero
p.	RsvdP_31-29	RO-Zero

(The MR Enable field must be written with 0, then the NumVH field must be written with the value returned by the MaxVH field, and then the MR Enable field must be written with 1 before this register is tested.)

Note: Testing this requires programming the designated Memory BAR and enabling Memory space. The value of [table] is the BAR contents (pointed to by the Function Table BIR field value) plus the offset (value in the Function Table Offset field multiplied by 8). The value of [size] is the value in the Function Table Entry Size field multiplied by 4. All values of [n] between 0 and the value in the MaxVH field inclusive must be tested.

Function Table Entry (n) VC to VL Map 2 ([table] + ([n] * [size])) + 1Ch — DWORD
(Endpoint)

- | | | |
|----|---|---------|
| a. | VC4 VL Map | |
| | (if Is Main BF is 1 and MaxVL is non-zero and VC Capability Supported is 1 and Num VC Resources Hardware Present is 4 or greater) | |
| | after first writing VC4 VL Map Enable with 0 | RW |
| | after first writing VC4 VL Map Enable with 1 | RO |
| | (if Is Main BF is 1 and MaxVL is 000b) | RO-Zero |
| | (if Is Main BF is 1 and VC Capability Supported is 0) | RO-Zero |
| | (if Is Main BF is 1 and Num VC Resources Hardware Present is less than 4) | RO-Zero |
| | (if Is Main BF is 0) | RO-Zero |
| b. | RsvdP_3 | RO-Zero |
| c. | VC4 VL Map Enable | |
| | (if Is Main BF is 1 and MaxVL is non-zero and VC Capability Supported is 1 and Num VC Resources Hardware Present is 4 or greater) | RW |
| | (if Is Main BF is 1 and MaxVL is 000b) | RO-Zero |
| | (if Is Main BF is 1 and VC Capability Supported is 0) | RO-Zero |
| | (if Is Main BF is 1 and Num VC Resources Hardware Present is less than 4) | RO-Zero |
| | (if Is Main BF is 0) | RO-Zero |
| d. | RsvdP_7-5 | RO-Zero |
| e. | VC5 VL Map | |
| | (if Is Main BF is 1 and MaxVL is non-zero and VC Capability Supported is 1 and Num VC Resources Hardware Present is 5 or greater) | |
| | after first writing VC5 VL Map Enable with 0 | RW |
| | after first writing VC5 VL Map Enable with 1 | RO |
| | (if Is Main BF is 1 and MaxVL is 000b) | RO-Zero |
| | (if Is Main BF is 1 and VC Capability Supported is 0) | RO-Zero |
| | (if Is Main BF is 1 and Num VC Resources Hardware Present is less than 5) | RO-Zero |
| | (if Is Main BF is 0) | RO-Zero |
| f. | RsvdP_11 | RO-Zero |
| g. | VC5 VL Map Enable | |
| | (if Is Main BF is 1 and MaxVL is non-zero and VC Capability Supported is 1 and Num VC Resources Hardware Present is 5 or greater) | RW |
| | (if Is Main BF is 1 and MaxVL is 000b) | RO-Zero |
| | (if Is Main BF is 1 and VC Capability Supported is 0) | RO-Zero |
| | (if Is Main BF is 1 and Num VC Resources Hardware Present is less than 5) | RO-Zero |
| | (if Is Main BF is 0) | RO-Zero |

h.	RsvdP_15-13	RO-Zero
i.	VC6 VL Map	
	(if Is Main BF is 1 and MaxVL is non-zero and VC Capability Supported is 1 and Num VC Resources Hardware Present is 6 or greater)	
	after first writing VC6 VL Map Enable with 0	RW
	after first writing VC6 VL Map Enable with 1	RO
	(if Is Main BF is 1 and MaxVL is 000b)	RO-Zero
	(if Is Main BF is 1 and VC Capability Supported is 0)	RO-Zero
	(if Is Main BF is 1 and Num VC Resources Hardware Present is less than 6)	RO-Zero
	(if Is Main BF is 0)	RO-Zero
j.	RsvdP_19	RO-Zero
k.	VC6 VL Map Enable	
	(if Is Main BF is 1 and MaxVL is non-zero and VC Capability Supported is 1 and Num VC Resources Hardware Present is 6 or greater)	RW
	(if Is Main BF is 1 and MaxVL is 000b)	RO-Zero
	(if Is Main BF is 1 and VC Capability Supported is 0)	RO-Zero
	(if Is Main BF is 1 and Num VC Resources Hardware Present is less than 6)	RO-Zero
	(if Is Main BF is 0)	RO-Zero
l.	RsvdP_23-21	RO-Zero
m.	VC7 VL Map	
	(if Is Main BF is 1 and MaxVL is non-zero and VC Capability Supported is 1 and Num VC Resources Hardware Present is 7)	
	after first writing VC7 VL Map Enable with 0	RW
	after first writing VC7 VL Map Enable with 1	RO
	(if Is Main BF is 1 and MaxVL is 000b)	RO-Zero
	(if Is Main BF is 1 and VC Capability Supported is 0)	RO-Zero
	(if Is Main BF is 1 and Num VC Resources Hardware Present is less than 7)	RO-Zero
	(if Is Main BF is 0)	RO-Zero
n.	RsvdP_27	RO-Zero
o.	VC7 VL Map Enable	
	(if Is Main BF is 1 and MaxVL is non-zero and VC Capability Supported is 1 and Num VC Resources Hardware Present is 7)	RW
	(if Is Main BF is 1 and MaxVL is 000b)	RO-Zero
	(if Is Main BF is 1 and VC Capability Supported is 0)	RO-Zero
	(if Is Main BF is 1 and Num VC Resources Hardware Present is less than 7)	RO-Zero
	(if Is Main BF is 0)	RO-Zero
p.	RsvdP_31-29	RO-Zero

(The MR Enable field must be written with 0, then the NumVH field must be written with the value returned by the MaxVH field, and then the MR Enable field must be written with 1 before this register is tested.)

Note: Testing this requires programming the designated Memory BAR and enabling Memory space. The value of [table] is the BAR contents (pointed to by the Function Table BIR field value) plus the offset (value in the Function Table Offset field multiplied by 8). The value of

[size] is the value in the Function Table Entry Size field multiplied by 4. All values of [n] between 0 and the value in the MaxVH field inclusive must be tested.

Function Table Entry (n) Reserved ([table] + ([n] * [size])) + 20h — 4 DWORDS

(Endpoint)

- | | |
|-------------------------------|---------|
| a. Reserved register (1st DW) | RO-Zero |
| b. Reserved register (2nd DW) | RO-Zero |
| c. Reserved register (3rd DW) | RO-Zero |
| d. Reserved register (4th DW) | RO-Zero |

(The MR Enable field must be written with 0, then the NumVH field must be written with the value returned by the MaxVH field, and then the MR Enable field must be written with 1 before this register is tested.)

Note: Testing this requires programming the designated Memory BAR and enabling Memory space. The value of [table] is the BAR contents (pointed to by the Function Table BIR field value) plus the offset (value in the Function Table Offset field multiplied by 8). The value of [size] is the value in the Function Table Entry Size field multiplied by 4. All values of [n] between 0 and the value in the MaxVH field inclusive must be tested.

Function Table Entry (n) VC Resource State (m) ([table] + ([n] * [size])) + 30h + ([m] * 4) — DWORD

(Endpoint)

- | | |
|---|---------|
| a. VC Enabled
(if [m] is 0) | RO-Ones |
| (if [m] is non-zero
and [m] is less than or equal to Num VC Resources Hardware Present
and [m] is less than or equal to VC Extended VC Count) | RO |
| (if [m] is nonzero
and [m] is less than or equal to Num VC Resources Hardware Present
and [m] is greater than VC Extended VC Count) | RO-Zero |
| b. VC Negotiation Pending
(if [m] is less than or equal to Num VC Resources Hardware Present
and [m] is less than or equal to VC Extended VC Count) | RO |
| (if [m] is less than or equal to Num VC Resources Hardware Present
and [m] is greater than VC Extended VC Count) | RO-Zero |
| c. RsvdZ_3-2 | RO-Zero |
| d. VC ID
(if [m] is 0) | RO-Zero |
| (if [m] is non-zero
and [m] is less than or equal to Num VC Resources Hardware Present
and [m] is less than or equal to VC Extended VC Count) | RO |
| (if [m] is nonzero
and [m] is less than or equal to Num VC Resources Hardware Present
and [m] is greater than VC Extended VC Count) | RO-Zero |

- | | |
|--|---------|
| e. RsvdZ_15-7 | RO-Zero |
| f. TC to VC Map | |
| (if [m] is 0) | RO |
| (if [m] is non-zero | |
| and [m] is less than or equal to Num VC Resources Hardware Present | |
| and [m] is less than or equal to VC Extended VC Count) | RO |
| (if [m] is nonzero | |
| and [m] is less than or equal to Num VC Resources Hardware Present | |
| and [m] is greater than VC Extended VC Count) | RO-Zero |
| g. RsvdZ_31-24 | RO-Zero |

(The MR Enable field must be written with 0, then the NumVH field must be written with the value returned by the MaxVH field, and then the MR Enable field must be written with 1 before this register is tested. The register group is tested twice if the Num VC Resources Hardware Present field is greater than 0. The first time it is tested after the VC Extended VC Count field is written with the value returned in the Num VC Resources Hardware Present field. The second time it is tested after the VC Extended VC Count field is written with 000b, but only if the Num VC Resources Hardware Present field is greater than 0.)

Note: Testing this requires programming the designated Memory BAR and enabling Memory space. The value of [table] is the BAR contents (pointed to by the Function Table BIR field value) plus the offset (value in the Function Table Offset field multiplied by 8). The value of [size] is the value in the Function Table Entry Size field multiplied by 4. All values of [n] between 0 and the value in the MaxVH field inclusive must be tested. All values of [m] between 0 and the value in the Num VC Resources Hardware Present field inclusive must be tested.

Function Table Entry (n) MFVC Resource State (m) ([table] + ([n] * [size])) + 30h + [offset] + ([m] * 4) — DWORD

(Endpoint)

- | | |
|--|---------|
| a. VC Enabled | |
| (if [m] is 0) | RO-Ones |
| (if [m] is non-zero | |
| and [m] is less than or equal to Num MFVC Resources Hardware Present | |
| and [m] is less than or equal to MFVC Extended VC Count) | RO |
| (if [m] is nonzero | |
| and [m] is less than or equal to Num MFVC Resources Hardware Present | |
| and [m] is greater than MFVC Extended VC Count) | RO-Zero |
| b. VC Negotiation Pending | |
| (if [m] is less than or equal to Num MFVC Resources Hardware Present | |
| and [m] is less than or equal to MFVC Extended VC Count) | RO |
| (if [m] is less than or equal to Num MFVC Resources Hardware Present | |
| and [m] is greater than MFVC Extended VC Count) | RO-Zero |
| c. RsvdZ_3-2 | RO-Zero |

- | | |
|--|---------|
| d. VC ID | |
| (if [m] is 0) | RO-Zero |
| (if [m] is non-zero | |
| and [m] is less than or equal to Num MFVC Resources Hardware Present | |
| and [m] is less than or equal to MFVC Extended VC Count) | RO |
| (if [m] is nonzero | |
| and [m] is less than or equal to Num MFVC Resources Hardware Present | |
| and [m] is greater than MFVC Extended VC Count) | RO-Zero |
| e. RsvdZ_15-7 | RO-Zero |
| f. TC to VC Map | |
| (if [m] is 0) | RO |
| (if [m] is non-zero | |
| and [m] is less than or equal to Num MFVC Resources Hardware Present | |
| and [m] is less than or equal to MFVC Extended VC Count) | RO |
| (if [m] is nonzero | |
| and [m] is less than or equal to Num MFVC Resources Hardware Present | |
| and [m] is greater than MFVC Extended VC Count) | RO-Zero |
| g. RsvdZ_31-24 | RO-Zero |

(This group of registers only exists if the MFVC Capability Supported field is a 1, and therefore this group of registers is not tested if this condition is not true. The MR Enable field must be written with 0, then the NumVH field must be written with the value returned by the MaxVH field, and then the MR Enable field must be written with 1 before this register is tested. If this register group is tested at all, it is tested twice if the Num VC Resources Hardware Present field is greater than 0. The first time it is tested after the MFVC Extended VC Count field is written with the value returned in the Num MFVC Resources Hardware Present field. The second time it is tested after the MFVC Extended VC Count field is written with 000b, but only if the Num VC Resources Hardware Present field is greater than 0.)

Note: Testing this requires programming the designated Memory BAR and enabling Memory space. The value of [table] is the BAR contents (pointed to by the Function Table BIR field value) plus the offset (value in the Function Table Offset field multiplied by 8). The value of [size] is the value in the Function Table Entry Size field multiplied by 4. All values of [n] between 0 and the value in the MaxVH field inclusive must be tested. If the MFVC Capability Supported field is a 1, then all values of [m] between 0 and the value in the Num MFVC Resources Hardware Present field inclusive must be tested. The value of [offset] is the value returned by the Num VC Resources Present field plus one and then multiplied by 4.

Function Table Entry (n) Reserved ([table] + ([n] * [size])) + 30h + [offset] + ([m] * 4) — [r] DWORDS

(Endpoint)

(all bits)

RO-Zero

(This group of registers only exists if [r] is larger than 0, and therefore this group of registers is not tested if this condition is not true. The MR Enable field must be written with 0, then the NumVH field must be written with the value returned by the MaxVH field, and then the MR Enable field must be written with 1 before this register is tested.)

Note: Testing this requires programming the designated Memory BAR and enabling Memory space. The value of [table] is the BAR contents (pointed to by the Function Table BIR field value) plus the offset (value in the Function Table Offset field multiplied by 8). The value of [size] is the value in the Function Table Entry Size field multiplied by 4. The value of [r] is $((([Function\ Table\ Entry\ Size] * 4) - ((12 * 4) + [offset])) / 4)$. All values of [n] between 0 and the value in the MaxVH field inclusive must be tested. All values of [m] between 0 and $((([Function\ Table\ Entry\ Size] * 4) - ((12 * 4) + [offset])) / 4) - 1$ must be tested. The value of [offset] is the $([Num\ VC\ Resources\ Present] + 1) * 4 + \{only\ if\ the\ MFVC\ Capability\ Supported\ field\ is\ a\ 1: ([Num\ MFVC\ Resources\ Present] + 1) * 4\}$.

Port Interrupt Status Bitmap ([table] - 8) — 8 DWORDS

(Switch)

- a. (bits [Num Port Table Entries]-1 to 0)
 - (for bit [n] if Port Table Entry [n] Port Present is 1 and VS is Authorized) RO
 - Note: All values of [n] from 0 to [Num Port Table Entries]-1 are tested.
 - (for bit [n] if Port Table Entry [n] Port Present is 0) RO-Zero
 - Note: All values of [n] from 0 to [Num Port Table Entries]-1 are tested.
 - (if VS is not Authorized) RO-Zero
- b. (bits (8 * 32) - 1 to [Num Port Table Entries]) RO-Zero

(The MR Enable field must be written with 0, then the NumVH field must be written with the value returned by the MaxVH field, and then the MR Enable field must be written with 1 before this register is tested.)

Note: Testing this requires programming the designated Memory BAR and enabling Memory space. The value of [table] is the BAR contents (pointed to by the Port Table BIR field value) plus the offset (value in the Port Table Offset field multiplied by 8).

Port Table Entry (n) Port Capability ([table] + ([n] * [size])) + 00h — DWORD

(Switch)

- a. Port Present
 - (if VS is Authorized) RO
 - (if VS is not Authorized) RO-Zero
- b. Non-PCIe Port (Management Port)
 - (if VS is Authorized) RO
 - (if VS is not Authorized) RO-Zero
- c. Link MR Capable
 - (if VS is Authorized and Non-PCIe Port (Management Port) is 0) RO
 - (if VS is Authorized and Non-PCIe Port (Management Port) is 1) RO-Zero
 - (if VS is not Authorized) RO-Zero

- | | |
|---|---------|
| d. Link Direction Supported | |
| (if VS is Authorized and Non-PCIe Port (Management Port) is 0) | RO |
| (if VS is Authorized and Non-PCIe Port (Management Port) is 1) | RO |
| (if VS is not Authorized) | RO-Zero |
| e. Port Direction Supported | |
| (if VS is Authorized and Non-PCIe Port (Management Port) is 0) | RO |
| (if VS is Authorized and Non-PCIe Port (Management Port) is 1) | RO |
| (if VS is not Authorized) | RO-Zero |
| f. RsvdZ_15-8 | RO-Zero |
| g. MaxVH | |
| (if VS is Authorized and Non-PCIe Port (Management Port) is 0 and Link MR Capable is 1) | RO |
| (if VS is Authorized and Non-PCIe Port (Management Port) is 0 and Link MR Capable is 0) | RO-Zero |
| (if VS is Authorized and Non-PCIe Port (Management Port) is 1) | RO-Zero |
| (if VS is not Authorized) | RO-Zero |
| h. PCIe Offset | |
| (if VS is Authorized and Non-PCIe Port (Management Port) is 0) | RO |
| (if VS is Authorized and Non-PCIe Port (Management Port) is 1) | RO-Zero |
| (if VS is not Authorized) | RO-Zero |

(The MR Enable field must be written with 0, then the NumVH field must be written with the value returned by the MaxVH field, and then the MR Enable field must be written with 1 before this register is tested.)

Note: Testing this requires programming the designated Memory BAR and enabling Memory space. The value of [table] is the BAR contents (pointed to by the Port Table BIR field value) plus the offset (value in the Port Table Offset field multiplied by 8). The value of [size] is the value in the Port Table Entry Size field multiplied by 4. All values of [n] between 0 and the value in the Num Port Table Entries field minus one inclusive must be tested.

Port Table Entry (n) Port Control 1 ([table] + ([n] * [size])) + 04h — DWORD
(Switch)

- | | |
|--|---------|
| a. Port DL_Up Interrupt Enable | |
| (if VS is Authorized and Non-PCIe Port (Management Port) is 0) | RW |
| (if VS is Authorized and Non-PCIe Port (Management Port) is 1) | RO-Zero |
| (if VS is not Authorized) | RO-Zero |
| b. Port DL_Down Interrupt Enable | |
| (if VS is Authorized and Non-PCIe Port (Management Port) is 0) | RW |
| (if VS is Authorized and Non-PCIe Port (Management Port) is 1) | RO-Zero |
| (if VS is not Authorized) | RO-Zero |
| c. Port PCIe Capability Interrupt Enable | |
| (if VS is Authorized and Non-PCIe Port (Management Port) is 0) | RW |
| (if VS is Authorized and Non-PCIe Port (Management Port) is 1) | RO-Zero |
| (if VS is not Authorized) | RO-Zero |
| d. Link Retrain Interrupt Enable | |
| (if VS is Authorized and Non-PCIe Port (Management Port) is 0) | RW |
| (if VS is Authorized and Non-PCIe Port (Management Port) is 1) | RO-Zero |
| (if VS is not Authorized) | RO-Zero |

e.	Beacon/WAKE# Interrupt Enable (if VS is Authorized and Non-PCIe Port (Management Port) is 0)	RW or RO-Zero
	(if VS is Authorized and Non-PCIe Port (Management Port) is 1)	RO-Zero
	(if VS is not Authorized)	RO-Zero
f.	MR Uncorrectable Fatal TLP Error Interrupt Enable (if VS is Authorized and Non-PCIe Port (Management Port) is 0)	RW
	(if VS is Authorized and Non-PCIe Port (Management Port) is 1)	RO-Zero
	(if VS is not Authorized)	RO-Zero
g.	RsvdP_6	RO-Zero
h.	MR Uncorrectable Global Key Error Interrupt Enable (if VS is Authorized and Non-PCIe Port (Management Port) is 0)	RW
	(if VS is Authorized and Non-PCIe Port (Management Port) is 1)	RO-Zero
	(if VS is not Authorized)	RO-Zero
i.	MR Correctable Global Key Error Interrupt Enable (if VS is Authorized and Non-PCIe Port (Management Port) is 0 and VS Global Key Entering Check Supported is 1)	RW
	(if VS is Authorized and Non-PCIe Port (Management Port) is 0 and VS Global Key Exiting Check Supported is 1)	RW
	(if VS is Authorized and Non-PCIe Port (Management Port) is 0 and VS Global Key Entering Check Supported is 0 and VS Global Key Exiting Check Supported is 0)	RO-Zero
	(if VS is Authorized and Non-PCIe Port (Management Port) is 1)	RO-Zero
	(if VS is not Authorized)	RO-Zero
j.	RsvdP_9	RO-Zero
k.	Physical Hot-Plug Interrupt Enable (if VS is Authorized and Non-PCIe Port (Management Port) is 0)	RW
	(if VS is Authorized and Non-PCIe Port (Management Port) is 1)	RO-Zero
	(if VS is not Authorized)	RO-Zero
l.	RsvdP_15-11	RO-Zero
m.	NumVH (if VS is Authorized and Non-PCIe Port (Management Port) is 0)	RW
	after first writing MR Enable with 0	RO
	after first writing MR Enable with 1	RO-Zero
	(if VS is Authorized and Non-PCIe Port (Management Port) is 1)	RO-Zero
	(if VS is not Authorized)	RO-Zero
n.	RsvdP_30-24	RO-Zero
o.	MR Enable (if VS is Authorized and Non-PCIe Port (Management Port) is 0)	RW
	(if VS is Authorized and Non-PCIe Port (Management Port) is 1)	RO-Zero
	(if VS is not Authorized)	RO-Zero

(The MR Enable field must be written with 0, then the NumVH field must be written with the value returned by the MaxVH field, and then the MR Enable field must be written with 1 before this register is tested.)

Note: Testing this requires programming the designated Memory BAR and enabling Memory space. The value of [table] is the BAR contents (pointed to by the Port Table BIR field value) plus the offset (value in the Port Table Offset field multiplied by 8). The value of [size] is the

value in the Port Table Entry Size field multiplied by 4. All values of [n] between 0 and the value in the Num Port Table Entries field minus one inclusive must be tested.

Port Table Entry (n) Port Control 2 ([table] + ([n] * [size])) + 08h — DWORD

(Switch)

- | | |
|--|---------|
| a. Port Enable | |
| (if VS is Authorized and Non-PCIe Port (Management Port) is 0) | RW |
| (if VS is Authorized and Non-PCIe Port (Management Port) is 1) | RO-Zero |
| (if VS is not Authorized) | RO-Zero |
| b. Link MR-IOV Enable | |
| (if VS is Authorized and Non-PCIe Port (Management Port) is 0 and Link MR Capable is 1) | RW |
| (if VS is Authorized and Non-PCIe Port (Management Port) is 0 and Link MR Capable is 0) | RO-Zero |
| (if VS is Authorized and Non-PCIe Port (Management Port) is 1) | RO-Zero |
| (if VS is not Authorized) | RO-Zero |
| c. RsvdP_7-2 | RO-Zero |
| d. Link Direction Control | |
| (if VS is Authorized and Non-PCIe Port (Management Port) is 0) | RW |
| (only those values supported in Link Direction Supported are written) | |
| (if VS is Authorized and Non-PCIe Port (Management Port) is 1) | RO-Zero |
| (if VS is not Authorized) | RO-Zero |
| e. RsvdP_11-10 | RO-Zero |
| f. PM_PME Triggers Beacon/WAKE# | |
| (if VS is Authorized and Non-PCIe Port (Management Port) is 0) | RW |
| (if VS is Authorized and Non-PCIe Port (Management Port) is 1) | RO-Zero |
| (if VS is not Authorized) | RO-Zero |
| g. Send PME_Enter_L23 DLLP | RO-Zero |
| h. Port Direction Control | |
| (if VS is Authorized and Non-PCIe Port (Management Port) is 0 and Port Direction Supported is 11b) | RW |
| (if VS is Authorized and Non-PCIe Port (Management Port) is 0 and Port Direction Supported is 10b) | RO-Ones |
| (if VS is Authorized and Non-PCIe Port (Management Port) is 0 and Port Direction Supported is 01b) | RO-Zero |
| (if VS is Authorized and Non-PCIe Port (Management Port) is 1) | RO-Zero |
| (if VS is not Authorized) | RO-Zero |
| i. RsvdP_15 | RO-Zero |
| j. VL Enable | |
| (if VS is Authorized and Non-PCIe Port (Management Port) is 0) | |
| (for bit 0) | RO-Ones |
| (for bits [MaxVL] to 1) | RW |
| (for bits 7 to [MaxVL] + 1) | RO-Zero |
| (if VS is Authorized and Non-PCIe Port (Management Port) is 1) | RO-Zero |
| (if VS is not Authorized) | RO-Zero |

k. RsvdP_31-24

RO-Zero

(The MR Enable field must be written with 0, then the NumVH field must be written with the value returned by the MaxVH field, and then the MR Enable field must be written with 1 before this register is tested.)

Note: Testing this requires programming the designated Memory BAR and enabling Memory space. The value of [table] is the BAR contents (pointed to by the Port Table BIR field value) plus the offset (value in the Port Table Offset field multiplied by 8). The value of [size] is the value in the Port Table Entry Size field multiplied by 4. All values of [n] between 0 and the value in the Num Port Table Entries field minus one inclusive must be tested.

Port Table Entry (n) Port Status ([table] + ([n] * [size])) + 0Ch — DWORD

(Switch)

- | | |
|--|---------|
| a. Port DL_Up Interrupt Status | |
| (if VS is Authorized and Non-PCIe Port (Management Port) is 0) | RW1C |
| (if VS is Authorized and Non-PCIe Port (Management Port) is 1) | RO-Zero |
| (if VS is not Authorized) | RO-Zero |
| b. Port DL_Down Interrupt Status | |
| (if VS is Authorized and Non-PCIe Port (Management Port) is 0) | RW1C |
| (if VS is Authorized and Non-PCIe Port (Management Port) is 1) | RO-Zero |
| (if VS is not Authorized) | RO-Zero |
| c. Port PCIe Capability Interrupt Status | |
| (if VS is Authorized and Non-PCIe Port (Management Port) is 0) | RW1C |
| (if VS is Authorized and Non-PCIe Port (Management Port) is 1) | RO-Zero |
| (if VS is not Authorized) | RO-Zero |
| d. Link Retrain Interrupt Status | |
| (if VS is Authorized and Non-PCIe Port (Management Port) is 0) | RW1C |
| (if VS is Authorized and Non-PCIe Port (Management Port) is 1) | RO-Zero |
| (if VS is not Authorized) | RO-Zero |
| e. Beacon/WAKE# Interrupt Status | |
| (if VS is Authorized and Non-PCIe Port (Management Port) is 0) | RW1C or |
| | RO-Zero |
| (if VS is Authorized and Non-PCIe Port (Management Port) is 1) | RO-Zero |
| (if VS is not Authorized) | RO-Zero |
| f. RsvdZ_9-5 | RO-Zero |
| g. Physical Hot-Plug Interrupt Status | |
| (if VS is Authorized and Non-PCIe Port (Management Port) is 0) | RW1C |
| (if VS is Authorized and Non-PCIe Port (Management Port) is 1) | RO-Zero |
| (if VS is not Authorized) | RO-Zero |
| h. RsvdZ_15-11 | RO-Zero |
| i. VL Negotiation Pending | |
| (if VS is Authorized and Non-PCIe Port (Management Port) is 0) | RO |
| (if VS is Authorized and Non-PCIe Port (Management Port) is 1) | RO-Zero |
| (if VS is not Authorized) | RO-Zero |
| j. RsvdZ_31-24 | RO-Zero |

(The MR Enable field must be written with 0, then the NumVH field must be written with the value returned by the MaxVH field, and then the MR Enable field must be written with 1 before this register is tested.)

Note: Testing this requires programming the designated Memory BAR and enabling Memory space. The value of [table] is the BAR contents (pointed to by the Port Table BIR field value) plus the offset (value in the Port Table Offset field multiplied by 8). The value of [size] is the value in the Port Table Entry Size field multiplied by 4. All values of [n] between 0 and the value in the Num Port Table Entries field minus one inclusive must be tested.

Port Table Entry (n) Link Partner Training Status ([table] + ([n] * [size])) + 10h — DWORD

(Switch)

- | | |
|--|---------|
| a. Link Partner Detected | |
| (if VS is Authorized and Non-PCIe Port (Management Port) is 0) | RO |
| (if VS is Authorized and Non-PCIe Port (Management Port) is 1) | RO-Zero |
| (if VS is not Authorized) | RO-Zero |
| b. Link Direction Status | |
| (if VS is Authorized and Non-PCIe Port (Management Port) is 0) | RO |
| (if VS is Authorized and Non-PCIe Port (Management Port) is 1) | RO-Zero |
| (if VS is not Authorized) | RO-Zero |
| c. Link Partner is MR | |
| (if VS is Authorized and Non-PCIe Port (Management Port) is 0) | RO |
| (if VS is Authorized and Non-PCIe Port (Management Port) is 1) | RO-Zero |
| (if VS is not Authorized) | RO-Zero |
| d. RsvdZ_7-4 | RO-Zero |
| e. Link Partner MaxVH | |
| (if VS is Authorized and Non-PCIe Port (Management Port) is 0) | RO |
| (if VS is Authorized and Non-PCIe Port (Management Port) is 1) | RO-Zero |
| (if VS is not Authorized) | RO-Zero |
| f. Link Partner MaxVL | |
| (if VS is Authorized and Non-PCIe Port (Management Port) is 0) | RO |
| (if VS is Authorized and Non-PCIe Port (Management Port) is 1) | RO-Zero |
| (if VS is not Authorized) | RO-Zero |
| g. RsvdZ_19 | RO-Zero |
| h. Link Partner Protocol Version | |
| (if VS is Authorized and Non-PCIe Port (Management Port) is 0) | RO |
| (if VS is Authorized and Non-PCIe Port (Management Port) is 1) | RO-Zero |
| (if VS is not Authorized) | RO-Zero |
| i. Link Partner was Authorized | |
| (if VS is Authorized and Non-PCIe Port (Management Port) is 0) | RO |
| (if VS is Authorized and Non-PCIe Port (Management Port) is 1) | RO-Zero |
| (if VS is not Authorized) | RO-Zero |
| j. Link Partner Device/Port Type | |
| (if VS is Authorized and Non-PCIe Port (Management Port) is 0) | RO |
| (if VS is Authorized and Non-PCIe Port (Management Port) is 1) | RO-Zero |
| (if VS is not Authorized) | RO-Zero |
| k. Link Partner Mixed Device/Port Type | |
| (if VS is Authorized and Non-PCIe Port (Management Port) is 0) | RO |
| (if VS is Authorized and Non-PCIe Port (Management Port) is 1) | RO-Zero |
| (if VS is not Authorized) | RO-Zero |
| l. RsvdZ_29 | RO-Zero |

- | | |
|--|---------|
| m. Link Partner VH FC | |
| (if VS is Authorized and Non-PCIe Port (Management Port) is 0) | RO |
| (if VS is Authorized and Non-PCIe Port (Management Port) is 1) | RO-Zero |
| (if VS is not Authorized) | RO-Zero |
| n. RsvdZ_31 | RO-Zero |

(The MR Enable field must be written with 0, then the NumVH field must be written with the value returned by the MaxVH field, and then the MR Enable field must be written with 1 before this register is tested.)

Note: Testing this requires programming the designated Memory BAR and enabling Memory space. The value of [table] is the BAR contents (pointed to by the Port Table BIR field value) plus the offset (value in the Port Table Offset field multiplied by 8). The value of [size] is the value in the Port Table Entry Size field multiplied by 4. All values of [n] between 0 and the value in the Num Port Table Entries field minus one inclusive must be tested.

**Port Table Entry (n) VL Arbitration Capability and Status ([table] + ([n] * [size])) + 14h -
- DWORD**

(Switch)

- | | |
|---|---------|
| a. VL Arbitration Capability | |
| (if VS is Authorized and Non-PCIe Port (Management Port) is 0
and MaxVL is non-zero) | |
| (bits 11, 5-0) | RO |
| (bits 10-6) | RO-Zero |
| (if VS is Authorized and Non-PCIe Port (Management Port) is 0
and MaxVL is 0) | RO-Zero |
| (if VS is Authorized and Non-PCIe Port (Management Port) is 1) | RO-Zero |
| (if VS is not Authorized) | RO-Zero |
| b. VL Arbitration Status | |
| (if VS is Authorized and Non-PCIe Port (Management Port) is 0) | RO |
| (if VS is Authorized and Non-PCIe Port (Management Port) is 1) | RO-Zero |
| (if VS is not Authorized) | RO-Zero |
| c. RsvdZ_13 | RO-Zero |
| d. Reference Clock | |
| (if VS is Authorized and Non-PCIe Port (Management Port) is 0) | RO-Zero |
| (if VS is Authorized and Non-PCIe Port (Management Port) is 1) | RO-Zero |
| (if VS is not Authorized) | RO-Zero |
| e. MaxVL | |
| (if VS is Authorized and Non-PCIe Port (Management Port) is 0) | RO |
| (if VS is Authorized and Non-PCIe Port (Management Port) is 1) | RO-Zero |
| (if VS is not Authorized) | RO-Zero |
| f. RsvdZ_23-19 | RO-Zero |
| g. Maximum Time Slots | |
| (if VS is Authorized and Non-PCIe Port (Management Port) is 0) | RO |
| (if VS is Authorized and Non-PCIe Port (Management Port) is 1) | RO-Zero |
| (if VS is not Authorized) | RO-Zero |
| h. RsvdZ_31 | RO-Zero |

(The MR Enable field must be written with 0, then the NumVH field must be written with the value returned by the MaxVH field, and then the MR Enable field must be written with 1 before this register is tested.)

Note: Testing this requires programming the designated Memory BAR and enabling Memory space. The value of [table] is the BAR contents (pointed to by the Port Table BIR field value) plus the offset (value in the Port Table Offset field multiplied by 8). The value of [size] is the value in the Port Table Entry Size field multiplied by 4. All values of [n] between 0 and the value in the Num Port Table Entries field minus one inclusive must be tested.

Port Table Entry (n) VL Arbitration Control ([table] + ([n] * [size])) + 18h — DWORD
(Switch)

- | | |
|--|---------|
| a. Load VL Arbitration Table | RO-Zero |
| b. RsvdP_7-1 | RO-Zero |
| c. VL Arbitration Select | |
| (if VS is Authorized and Non-PCIe Port (Management Port) is 0 and MaxVL is non-zero) | RW |
| (only values corresponding to those reported in VL Arbitration Capability are written) | |
| (if VS is Authorized and Non-PCIe Port (Management Port) is 0 and MaxVL is 0) | RO-Zero |
| (if VS is Authorized and Non-PCIe Port (Management Port) is 1) | RO-Zero |
| (if VS is not Authorized) | RO-Zero |
| d. RsvdP_15-12 | RO-Zero |
| e. VL Strict Priority Arbitration | |
| (if VS is Authorized and Non-PCIe Port (Management Port) is 0 and MaxVL is non-zero) | RW |
| (if VS is Authorized and Non-PCIe Port (Management Port) is 0 and MaxVL is 0) | RO-Zero |
| (if VS is Authorized and Non-PCIe Port (Management Port) is 1) | RO-Zero |
| (if VS is not Authorized) | RO-Zero |
| f. RsvdP_31-24 | RO-Zero |

(The MR Enable field must be written with 0, then the NumVH field must be written with the value returned by the MaxVH field, and then the MR Enable field must be written with 1 before this register is tested.)

Note: Testing this requires programming the designated Memory BAR and enabling Memory space. The value of [table] is the BAR contents (pointed to by the Port Table BIR field value) plus the offset (value in the Port Table Offset field multiplied by 8). The value of [size] is the value in the Port Table Entry Size field multiplied by 4. All values of [n] between 0 and the value in the Num Port Table Entries field minus one inclusive must be tested.

Port Table Entry (n) VL Arbitration Table Offset ([table] + ([n] * [size])) + 1Ch — DWORD

(Switch)

- | | |
|--|---------|
| a. RsvdZ_1-0 | RO-Zero |
| b. VL Arbitration Table Offset | |
| (if VS is Authorized and Non-PCIe Port (Management Port) is 0) | RO |
| (if VS is Authorized and Non-PCIe Port (Management Port) is 1) | RO-Zero |
| (if VS is not Authorized) | RO-Zero |

(The MR Enable field must be written with 0, then the NumVH field must be written with the value returned by the MaxVH field, and then the MR Enable field must be written with 1 before this register is tested.)

Note: Testing this requires programming the designated Memory BAR and enabling Memory space. The value of [table] is the BAR contents (pointed to by the Port Table BIR field value) plus the offset (value in the Port Table Offset field multiplied by 8). The value of [size] is the value in the Port Table Entry Size field multiplied by 4. All values of [n] between 0 and the value in the Num Port Table Entries field minus one inclusive must be tested.

Port Table Entry (n) MR Error Status ([table] + ([n] * [size])) + 20h — WORD

(Switch)

- | | |
|---|---------|
| a. MR First Error Pointer | |
| (if VS is Authorized and Non-PCIe Port (Management Port) is 0) | ROS |
| (if VS is Authorized and Non-PCIe Port (Management Port) is 1) | RO-Zero |
| (if VS is not Authorized) | RO-Zero |
| b. MR Uncorrectable Fatal TLP Error Status | |
| (if VS is Authorized and Non-PCIe Port (Management Port) is 0) | RW1CS |
| (if VS is Authorized and Non-PCIe Port (Management Port) is 1) | RO-Zero |
| (if VS is not Authorized) | RO-Zero |
| c. RsvdZ_5 | RO-Zero |
| d. MR Uncorrectable Global Key Error Status | |
| (if VS is Authorized and Non-PCIe Port (Management Port) is 0) | RW1CS |
| (if VS is Authorized and Non-PCIe Port (Management Port) is 1) | RO-Zero |
| (if VS is not Authorized) | RO-Zero |
| e. MR Correctable Global Key Error Status | |
| (if VS is Authorized and Non-PCIe Port (Management Port) is 0 and VS Global Key Entering Check Supported is 1) | RW1CS |
| (if VS is Authorized and Non-PCIe Port (Management Port) is 0 and VS Global Key Exiting Check Supported is 1) | RW1CS |
| (if VS is Authorized and Non-PCIe Port (Management Port) is 0 and VS Global Key Entering Check Supported is 0 and VS Global Key Exiting Check Supported is 0) | RO-Zero |
| (if VS is Authorized and Non-PCIe Port (Management Port) is 1) | RO-Zero |
| (if VS is not Authorized) | RO-Zero |
| f. RsvdZ_14-8 | RO-Zero |

- | | |
|--|---------|
| g. MR Multiple Uncorrectable Error | |
| (if VS is Authorized and Non-PCIe Port (Management Port) is 0) | RW1CS |
| (if VS is Authorized and Non-PCIe Port (Management Port) is 1) | RO-Zero |
| (if VS is not Authorized) | RO-Zero |

(The MR Enable field must be written with 0, then the NumVH field must be written with the value returned by the MaxVH field, and then the MR Enable field must be written with 1 before this register is tested.)

Note: Testing this requires programming the designated Memory BAR and enabling Memory space. The value of [table] is the BAR contents (pointed to by the Port Table BIR field value) plus the offset (value in the Port Table Offset field multiplied by 8). The value of [size] is the value in the Port Table Entry Size field multiplied by 4. All values of [n] between 0 and the value in the Num Port Table Entries field minus one inclusive must be tested.

Port Table Entry (n) MR Error Control ([table] + ([n] * [size])) + 22h — WORD

(Switch)

- | | |
|--|---------|
| a. RsvdP_3-0 | RO-Zero |
| b. MR Uncorrectable Fatal TLP Error Mask | |
| (if VS is Authorized and Non-PCIe Port (Management Port) is 0) | RWS |
| (if VS is Authorized and Non-PCIe Port (Management Port) is 1) | RO-Zero |
| (if VS is not Authorized) | RO-Zero |
| c. RsvdP_5 | RO-Zero |
| d. MR Uncorrectable Global Key Error Mask | |
| (if VS is Authorized and Non-PCIe Port (Management Port) is 0) | RWS |
| (if VS is Authorized and Non-PCIe Port (Management Port) is 1) | RO-Zero |
| (if VS is not Authorized) | RO-Zero |
| e. MR Correctable Global Key Error Mask | |
| (if VS is Authorized and Non-PCIe Port (Management Port) is 0 | |
| and VS Global Key Entering Check Supported is 1) | RWS |
| (if VS is Authorized and Non-PCIe Port (Management Port) is 0 | |
| and VS Global Key Exiting Check Supported is 1) | RWS |
| (if VS is Authorized and Non-PCIe Port (Management Port) is 0 | |
| and VS Global Key Entering Check Supported is 0 | |
| and VS Global Key Exiting Check Supported is 0) | RO-Zero |
| (if VS is Authorized and Non-PCIe Port (Management Port) is 1) | RO-Zero |
| (if VS is not Authorized) | RO-Zero |
| f. RsvdP_15-8 | RO-Zero |

(The MR Enable field must be written with 0, then the NumVH field must be written with the value returned by the MaxVH field, and then the MR Enable field must be written with 1 before this register is tested.)

Note: Testing this requires programming the designated Memory BAR and enabling Memory space. The value of [table] is the BAR contents (pointed to by the Port Table BIR field value) plus the offset (value in the Port Table Offset field multiplied by 8). The value of [size] is the value in the Port Table Entry Size field multiplied by 4. All values of [n] between 0 and the value in the Num Port Table Entries field minus one inclusive must be tested.

Port Table Entry (n) Header Log Register ([table] + ([n] * [size])) + 24h — 9 DWORDS
(Switch)

a. Header Log register (1st DW)	
(if VS is Authorized and Non-PCIe Port (Management Port) is 0)	ROS
(if VS is Authorized and Non-PCIe Port (Management Port) is 1)	RO-Zero
(if VS is not Authorized)	RO-Zero
b. Header Log register (2nd DW)	
(if VS is Authorized and Non-PCIe Port (Management Port) is 0)	ROS
(if VS is Authorized and Non-PCIe Port (Management Port) is 1)	RO-Zero
(if VS is not Authorized)	RO-Zero
c. Header Log register (3rd DW)	
(if VS is Authorized and Non-PCIe Port (Management Port) is 0)	ROS
(if VS is Authorized and Non-PCIe Port (Management Port) is 1)	RO-Zero
(if VS is not Authorized)	RO-Zero
d. Header Log register (4th DW)	
(if VS is Authorized and Non-PCIe Port (Management Port) is 0)	ROS
(if VS is Authorized and Non-PCIe Port (Management Port) is 1)	RO-Zero
(if VS is not Authorized)	RO-Zero
e. Header Log register (5th DW)	
(if VS is Authorized and Non-PCIe Port (Management Port) is 0)	ROS
(if VS is Authorized and Non-PCIe Port (Management Port) is 1)	RO-Zero
(if VS is not Authorized)	RO-Zero
f. Header Log register (6th DW)	
(if VS is Authorized and Non-PCIe Port (Management Port) is 0)	ROS
(if VS is Authorized and Non-PCIe Port (Management Port) is 1)	RO-Zero
(if VS is not Authorized)	RO-Zero
g. Header Log register (7th DW)	
(if VS is Authorized and Non-PCIe Port (Management Port) is 0)	ROS
(if VS is Authorized and Non-PCIe Port (Management Port) is 1)	RO-Zero
(if VS is not Authorized)	RO-Zero
h. Header Log register (8th DW)	
(if VS is Authorized and Non-PCIe Port (Management Port) is 0)	ROS
(if VS is Authorized and Non-PCIe Port (Management Port) is 1)	RO-Zero
(if VS is not Authorized)	RO-Zero
i. Header Log register (9th DW)	
(if VS is Authorized and Non-PCIe Port (Management Port) is 0)	ROS
(if VS is Authorized and Non-PCIe Port (Management Port) is 1)	RO-Zero
(if VS is not Authorized)	RO-Zero

(The MR Enable field must be written with 0, then the NumVH field must be written with the value returned by the MaxVH field, and then the MR Enable field must be written with 1 before this register is tested.)

Note: Testing this requires programming the designated Memory BAR and enabling Memory space. The value of [table] is the BAR contents (pointed to by the Port Table BIR field value) plus the offset (value in the Port Table Offset field multiplied by 8). The value of [size] is the value in the Port Table Entry Size field multiplied by 4. All values of [n] between 0 and the value in the Num Port Table Entries field minus one inclusive must be tested.

Port Table Entry (n) Reserved ([table] + ([n] * [size])) + 48h + ([m] * 4) — [r] DWORDS

(Switch)

(all bits)

RO-Zero

(This group of registers only exists if [r] is larger than 0, and therefore this group of registers is not tested if this condition is not true. The MR Enable field must be written with 0, then the NumVH field must be written with the value returned by the MaxVH field, and then the MR Enable field must be written with 1 before this register is tested.)

Note: Testing this requires programming the designated Memory BAR and enabling Memory space. The value of [table] is the BAR contents (pointed to by the Port Table BIR field value) plus the offset (value in the Port Table Offset field multiplied by 8). The value of [size] is the value in the Port Table Entry Size field multiplied by 4. The value of [r] is $([\text{offset}] - (19 * 4) / 4)$. All values of [n] between 0 and the value in the Num Port Table Entries field minus one inclusive must be tested. All values of [m] between 0 and $(([\text{offset}] - (19 * 4) / 4) - 1)$ must be tested. The value of [offset] is the value in the PCIe Offset field multiplied by 4.

Port Table Entry (n) PCI Bridge Control ([table] + ([n] * [size])) + [offset] + 00h — WORD

(Switch if PCIe Offset is non-zero)

a. Secondary Bus Reset

RW

b. RsvdP_15-1

RO-Zero

(The MR Enable field must be written with 0, then the NumVH field must be written with the value returned by the MaxVH field, and then the MR Enable field must be written with 1 before this register is tested.)

Note: Testing this requires programming the designated Memory BAR and enabling Memory space. The value of [table] is the BAR contents (pointed to by the Port Table BIR field value) plus the offset (value in the Port Table Offset field multiplied by 8). The value of [size] is the value in the Port Table Entry Size field multiplied by 4. All values of [n] between 0 and the value in the Num Port Table Entries field minus one inclusive must be tested. The value of [offset] is the value in the PCIe Offset field multiplied by 4. If the PCIe Offset field is zero, then this register does not exist and is not tested.

Port Table Entry (n) PCI Express Capabilities ([table] + ([n] * [size])) + [offset] + 02h — WORD

(Switch if PCIe Offset is non-zero)

a. Capability Version

RO

b. Device/Port Type

RO-Zero

c. Slot Implemented

(for non-FLR testing)

HwInit

(for FLR testing)

RO

d. Interrupt Message Number

RO-Zero

e. Undefined_14

RO

Note: This was once TCS Routing Supported.

f. RsvdP_15

RO-Zero

(The MR Enable field must be written with 0, then the NumVH field must be written with the value returned by the MaxVH field, and then the MR Enable field must be written with 1 before this register is tested.)

Note: Testing this requires programming the designated Memory BAR and enabling Memory space. The value of [table] is the BAR contents (pointed to by the Port Table BIR field value) plus the offset (value in the Port Table Offset field multiplied by 8). The value of [size] is the value in the Port Table Entry Size field multiplied by 4. All values of [n] between 0 and the value in the Num Port Table Entries field minus one inclusive must be tested. The value of [offset] is the value in the PCIe Offset field multiplied by 4. If the PCIe Offset field is zero, then this register does not exist and is not tested.

Port Table Entry (n) Device Capabilities ([table] + ([n] * [size])) + [offset] + 04h — DWORD

(Switch if PCIe Offset is non-zero)

- | | |
|--|---------|
| a. Max_Payload_Size Supported | RO |
| b. Phantom Functions Supported | RO-Zero |
| c. Extended Tag Field Supported | RO |
| d. Endpoint L0s Acceptable Latency | RO-Zero |
| e. Endpoint L1 Acceptable Latency | RO-Zero |
| f. Undefined_12 | RO |
| Note: This was once Attention Button Present. | |
| g. Undefined_13 | RO |
| Note: This was once Attention Indicator Present. | |
| h. Undefined_14 | RO |
| Note: This was once Power Indicator Present. | |
| i. Role-Based Error Reporting | RO-Ones |
| j. RsvdP_17-16 | RO-Zero |
| k. Captured Slot Power Limit Value | RO |
| l. Captured Slot Power Limit Scale | RO |
| m. Function Level Reset Capability | RO-Zero |
| n. RsvdP_31-29 | RO-Zero |

(The MR Enable field must be written with 0, then the NumVH field must be written with the value returned by the MaxVH field, and then the MR Enable field must be written with 1 before this register is tested.)

Note: Testing this requires programming the designated Memory BAR and enabling Memory space. The value of [table] is the BAR contents (pointed to by the Port Table BIR field value) plus the offset (value in the Port Table Offset field multiplied by 8). The value of [size] is the value in the Port Table Entry Size field multiplied by 4. All values of [n] between 0 and the value in the Num Port Table Entries field minus one inclusive must be tested. The value of [offset] is the value in the PCIe Offset field multiplied by 4. If the PCIe Offset field is zero, then this register does not exist and is not tested.

Port Table Entry (n) Device Control ([table] + ([n] * [size])) + [offset] + 08h — WORD

(Switch if PCIe Offset is non-zero)

- | | |
|---|---------|
| a. Correctable Error Reporting Enable | RO-Zero |
| b. Non-Fatal Error Reporting Enable | RO-Zero |
| c. Fatal Error Reporting Enable | RO-Zero |
| d. Unsupported Request Reporting Enable | RO-Zero |
| e. Enable Relaxed Ordering | RO-Zero |
| f. Max_Payload_Size | RO-Zero |
| g. Extended Tag Field Enable | RO-Zero |

h. Phantom Functions Enable	RO-Zero
i. Auxiliary (AUX) Power PM Enable (if Aux Current is non-zero in PM Capability) (if Aux Current is 000b in PM Capability)	RWS RWS or RO-Zero
j. Enable No Snoop	RO-Zero
k. Max_Read_Request_Size	RO-Zero
l. Bridge Configuration Retry Enable /Initiate Function Level Reset	RO-Zero

(The MR Enable field must be written with 0, then the NumVH field must be written with the value returned by the MaxVH field, and then the MR Enable field must be written with 1 before this register is tested.)

Note: Testing this requires programming the designated Memory BAR and enabling Memory space. The value of [table] is the BAR contents (pointed to by the Port Table BIR field value) plus the offset (value in the Port Table Offset field multiplied by 8). The value of [size] is the value in the Port Table Entry Size field multiplied by 4. All values of [n] between 0 and the value in the Num Port Table Entries field minus one inclusive must be tested. The value of [offset] is the value in the PCIe Offset field multiplied by 4. If the PCIe Offset field is zero, then this register does not exist and is not tested.

Port Table Entry (n) Device Status ([table] + ([n] * [size])) + [offset] + 0Ah — WORD

(Switch if PCIe Offset is non-zero)

a. Correctable Error Detected	RO-Zero
b. Non-Fatal Error Detected	RO-Zero
c. Fatal Error Detected	RO-Zero
d. Unsupported Request Detected	RO-Zero
e. AUX Power Detected	RO
f. Transactions Pending	RO-Zero
g. RsvdZ_15-6	RO-Zero

(The MR Enable field must be written with 0, then the NumVH field must be written with the value returned by the MaxVH field, and then the MR Enable field must be written with 1 before this register is tested.)

Note: Testing this requires programming the designated Memory BAR and enabling Memory space. The value of [table] is the BAR contents (pointed to by the Port Table BIR field value) plus the offset (value in the Port Table Offset field multiplied by 8). The value of [size] is the value in the Port Table Entry Size field multiplied by 4. All values of [n] between 0 and the value in the Num Port Table Entries field minus one inclusive must be tested. The value of [offset] is the value in the PCIe Offset field multiplied by 4. If the PCIe Offset field is zero, then this register does not exist and is not tested.

Port Table Entry (n) Link Capabilities ([table] + ([n] * [size])) + [offset] + 0Ch — DWORD

(Switch if PCIe Offset is non-zero)

a. Max Link Speed/Supported Link Speeds	RO
b. Maximum Link Width	RO
c. Active State Power Management (ASPM) Support	RO
d. L0s Exit Latency	RO
e. L1 Exit Latency	RO

- f. Clock Power Management
(if Port Direction Control is 1 (downstream port)) RO
(if Port Direction Control is 0 (upstream port)) RO-Zero
- g. Surprise Down Error Reporting Capable RO-Ones
- h. Data Link Layer Link Active Reporting Capable RO-Ones
- i. Link Bandwidth Notification Capability RO-Ones
- j. ASPM Optionality Compliance
(For Base 2.x or later testing)
(for non-FLR testing) HwInit
(for FLR testing) RO
- k. RsvdP_23
(For Base 2.x or later testing) RO-Zero
- l. RsvdP_23-22
(For Base 1.x testing) RO-Zero
- m. Port Number RO-Zero

(The MR Enable field must be written with 0, then the NumVH field must be written with the value returned by the MaxVH field, and then the MR Enable field must be written with 1 before this register is tested.)

Note: Testing this requires programming the designated Memory BAR and enabling Memory space. The value of [table] is the BAR contents (pointed to by the Port Table BIR field value) plus the offset (value in the Port Table Offset field multiplied by 8). The value of [size] is the value in the Port Table Entry Size field multiplied by 4. All values of [n] between 0 and the value in the Num Port Table Entries field minus one inclusive must be tested. The value of [offset] is the value in the PCIe Offset field multiplied by 4. If the PCIe Offset field is zero, then this register does not exist and is not tested.

Port Table Entry (n) Link Control ([table] + ([n] * [size])) + [offset] + 10h — WORD

(Switch if PCIe Offset is non-zero)

- a. Active State Power Management (ASPM) Control RW
- b. RsvdP_2 RO-Zero
- c. Read Completion Boundary (RCB) RO-Zero
- d. Link Disable
(if Link Direction Supported (bit 2) is 1) RW
(if Link Direction Supported (bit 2) is 0
and Port Direction Control is 1 (downstream port)) RW
(if Link Direction Supported (bit 2) is 0
and Port Direction Control is 0 (upstream port)) RO-Zero
- e. Retrain Link RO-Zero
- f. Common Clock Configuration RW
- g. Extended Synch RW
- h. Enable Clock Power Management
(if Port Direction Control is 1 (downstream port)) RO-Zero
(if Port Direction Control is 0 (upstream port)
and Port Table Entry Clock Power Management as 1) RW
- i. Hardware Autonomous Width Disable RW or
RO-Zero

- | | |
|---|---------------|
| j. Link Bandwidth Management Interrupt Enable
(if Port Direction Control is 1 (downstream port))
(if Port Direction Control is 0 (upstream port)) | RW
RO-Zero |
| k. Link Autonomous Bandwidth Interrupt Enable
(if Port Direction Control is 1 (downstream port))
(if Port Direction Control is 0 (upstream port)) | RW
RO-Zero |
| l. RsvdP_15-12 | RO-Zero |

(The MR Enable field must be written with 0, then the NumVH field must be written with the value returned by the MaxVH field, and then the MR Enable field must be written with 1 before this register is tested.)

Note: Testing this requires programming the designated Memory BAR and enabling Memory space. The value of [table] is the BAR contents (pointed to by the Port Table BIR field value) plus the offset (value in the Port Table Offset field multiplied by 8). The value of [size] is the value in the Port Table Entry Size field multiplied by 4. All values of [n] between 0 and the value in the Num Port Table Entries field minus one inclusive must be tested. The value of [offset] is the value in the PCIe Offset field multiplied by 4. If the PCIe Offset field is zero, then this register does not exist and is not tested.

Port Table Entry (n) Link Status ([table] + ([n] * [size])) + [offset] + 12h — WORD

(Switch if PCIe Offset is non-zero)

- | | |
|---|-----------------|
| a. Current Link Speed | RO |
| b. Negotiated Link Width | RO |
| c. Undefined_10 | RO |
| Note: This was once Link Training Error. | |
| d. Link Training
(if Port Direction Control is 1 (downstream port))
(if Port Direction Control is 0 (upstream port)) | RO
RO-Zero |
| e. Slot Clock Configuration
(for non-FLR testing)
(for FLR testing) | HwInit
RO |
| f. Data Link Layer Link Active
(if Port Direction Control is 1 (downstream port))
(if Port Direction Control is 0 (upstream port)) | RO
RO-Zero |
| g. Link Bandwidth Management Status
(if Port Direction Control is 1 (downstream port))
(if Port Direction Control is 0 (upstream port)) | RW1C
RO-Zero |
| h. Link Autonomous Bandwidth Status
(if Port Direction Control is 1 (downstream port))
(if Port Direction Control is 0 (upstream port)) | RW1C
RO-Zero |

(The MR Enable field must be written with 0, then the NumVH field must be written with the value returned by the MaxVH field, and then the MR Enable field must be written with 1 before this register is tested.)

Note: Testing this requires programming the designated Memory BAR and enabling Memory space. The value of [table] is the BAR contents (pointed to by the Port Table BIR field value) plus the offset (value in the Port Table Offset field multiplied by 8). The value of [size] is the value in the Port Table Entry Size field multiplied by 4. All values of [n] between 0 and the value in the Num Port Table Entries field minus one inclusive must be tested. The value of

[offset] is the value in the PCIe Offset field multiplied by 4. If the PCIe Offset field is zero, then this register does not exist and is not tested.

Port Table Entry (n) Slot Capabilities ([table] + ([n] * [size])) + [offset] + 14h — DWORD

(Switch if PCIe Offset is non-zero)

a. Attention Button Present (for non-FLR testing) (for FLR testing)	HwInit RO
b. Power Controller Present (for non-FLR testing) (for FLR testing)	HwInit RO
c. MRL Sensor Present (for non-FLR testing) (for FLR testing)	HwInit RO
d. Attention Indicator Present (for non-FLR testing) (for FLR testing)	HwInit RO
e. Power Indicator Present (for non-FLR testing) (for FLR testing)	HwInit RO
f. Hot-Plug Surprise (for non-FLR testing) (for FLR testing)	HwInit RO
g. Hot-Plug Capable (for non-FLR testing) (for FLR testing)	HwInit RO
h. Slot Power Limit Value (for non-FLR testing) (for FLR testing)	HwInit RO
i. Slot Power Limit Scale (for non-FLR testing) (for FLR testing)	HwInit RO
j. Electromechanical Interlock Present (for non-FLR testing) (for FLR testing)	HwInit RO
k. No Command Completed Support (for non-FLR testing) (for FLR testing)	HwInit RO
l. Physical Slot Number (for non-FLR testing) (for FLR testing)	HwInit RO

(The MR Enable field must be written with 0, then the NumVH field must be written with the value returned by the MaxVH field, and then the MR Enable field must be written with 1 before this register is tested.)

Note: Testing this requires programming the designated Memory BAR and enabling Memory space. The value of [table] is the BAR contents (pointed to by the Port Table BIR field value) plus the offset (value in the Port Table Offset field multiplied by 8). The value of [size] is the value in the Port Table Entry Size field multiplied by 4. All values of [n] between 0 and the

value in the Num Port Table Entries field minus one inclusive must be tested. The value of [offset] is the value in the PCIe Offset field multiplied by 4. If the PCIe Offset field is zero, then this register does not exist and is not tested. If Port Table Entry Slot Implemented is 0, then the entire register is tested as RO-Zero.

Port Table Entry (n) Slot Control ([table] + ([n] * [size])) + [offset] + 18h — WORD

(Switch if PCIe Offset is non-zero)

- | | | |
|----|--|------------------------|
| a. | Attention Button Pressed Enable
(if Port Table Entry Attention Button Present is 1)
(if Port Table Entry Attention Button Present is 0) | RW
RW or
RO-Zero |
| b. | Power Fault Detected Enable | RW or
RO-Zero |
| c. | MRL Sensor Changed Enable
(if Port Table Entry MRL Sensor Present is 1)
(if Port Table Entry MRL Sensor Present is 0) | RW
RW or
RO-Zero |
| d. | Presence Detect Changed Enable
(if Port Table Entry Hot-Plug Capable is 1)
(if Port Table Entry Hot-Plug Capable is 0) | RW
RW or
RO-Zero |
| e. | Command Completed Interrupt Enable
(if Port Table Entry No Command Completed Support is 0)
(if Port Table Entry No Command Completed Support is 1) | RW
RW or
RO-Zero |
| f. | Hot-Plug Interrupt Enable
(if Port Table Entry Hot-Plug Capable is 1)
(if Port Table Entry Hot-Plug Capable is 0) | RW
RW or
RO-Zero |
| g. | Attention Indicator Control
(if Port Table Entry Attention Indicator Present is 0) | RW or
RO-Zero |
| h. | Power Indicator Control
(if Port Table Entry Power Indicator Present is 0) | RW or
RO-Zero |
| i. | Power Controller Control
(if Port Table Entry Power Controller Present is 0) | RW or RO |
| j. | Electromechanical Interlock Control | RO-Zero |
| k. | Data Link Layer State Changed Enable
(if Port Table Entry Data Link Layer Link Active Reporting Capable is 1)
(if Port Table Entry Data Link Layer Link Active Reporting Capable is 0) | RW
RW or
RO-Zero |
| l. | RsvdP_15-13 | RO-Zero |

(The MR Enable field must be written with 0, then the NumVH field must be written with the value returned by the MaxVH field, and then the MR Enable field must be written with 1 before this register is tested.)

Note: Testing this requires programming the designated Memory BAR and enabling Memory space. The value of [table] is the BAR contents (pointed to by the Port Table BIR field value)

plus the offset (value in the Port Table Offset field multiplied by 8). The value of [size] is the value in the Port Table Entry Size field multiplied by 4. All values of [n] between 0 and the value in the Num Port Table Entries field minus one inclusive must be tested. The value of [offset] is the value in the PCIe Offset field multiplied by 4. If the PCIe Offset field is zero, then this register does not exist and is not tested. If Port Table Entry Slot Implemented is 0, then the entire register is tested as RO-Zero.

Port Table Entry (n) Slot Status ([table] + ([n] * [size])) + [offset] + 1Ah — WORD

(Switch if PCIe Offset is non-zero)

- | | |
|---|---------|
| a. Attention Button Pressed | |
| (if Port Table Entry Attention Button Present is 1) | RW1C |
| (if Port Table Entry Attention Button Present is 0) | RO-Zero |
| b. Power Fault Detected | |
| (if Port Table Entry Power Controller Present is 1) | RW1C |
| (if Port Table Entry Power Controller Present is 0) | RO-Zero |
| c. MRL Sensor Changed | |
| (if Port Table Entry MRL Sensor Present is 1) | RW1C |
| (if Port Table Entry MRL Sensor Present is 0) | RO-Zero |
| d. Presence Detect Changed | RW1C |
| e. Command Completed | |
| (if Port Table Entry No Command Completed Support is 0) | RW1C |
| (if Port Table Entry No Command Completed Support is 1) | RO-Zero |
| f. MRL Sensor State | RO |
| g. Presence Detect State | RO |
| h. Electromechanical Interlock Status | RO |
| i. Data Link Layer State Changed | RW1C |
| j. RsvdZ_15-9 | RO-Zero |

(The MR Enable field must be written with 0, then the NumVH field must be written with the value returned by the MaxVH field, and then the MR Enable field must be written with 1 before this register is tested.)

Note: Testing this requires programming the designated Memory BAR and enabling Memory space. The value of [table] is the BAR contents (pointed to by the Port Table BIR field value) plus the offset (value in the Port Table Offset field multiplied by 8). The value of [size] is the value in the Port Table Entry Size field multiplied by 4. All values of [n] between 0 and the value in the Num Port Table Entries field minus one inclusive must be tested. The value of [offset] is the value in the PCIe Offset field multiplied by 4. If the PCIe Offset field is zero, then this register does not exist and is not tested. If Port Table Entry Slot Implemented is 0, then the entire register is tested as RO-Zero.

Port Table Entry (n) Device Capabilities 2 ([table] + ([n] * [size])) + [offset] + 1Ch — DWORD

(Switch if PCIe Offset is non-zero)

- | | |
|---|---------|
| a. Completion Timeout Ranges Supported | RO-Zero |
| b. Completion Timeout Disable Supported | RO-Zero |
| c. ARI Forwarding Supported | RO-Zero |
| d. AtomicOp Routing Supported | |
| (For Base 2.x or later testing) | RO |

e.	32-bit AtomicOp Completer Supported (For Base 2.x or later testing)	RO-Zero
f.	64-bit AtomicOp Completer Supported (For Base 2.x or later testing)	RO-Zero
g.	128-bit CAS Completer Supported (For Base 2.x or later testing)	RO-Zero
h.	No RO-enabled PR-PR Passing (For Base 2.x or later testing) (for non-FLR testing) (for FLR testing)	HwInit RO
i.	LTR Mechanism Supported (For Base 2.x or later testing)	RO
j.	TPH Completer Supported (For Base 2.x or later testing)	RO-Zero
k.	RsvdP_17-14 (For Base 2.x or later testing)	RO-Zero
l.	OBFF Supported (For Base 2.x or later testing) (for non-FLR testing) (for FLR testing)	HwInit RO
m.	Extended Fmt Field Supported (For Base 2.x or later testing)	RO
n.	End-End TLP Prefix Supported (For Base 2.x or later testing)	RO
o.	Max End-End TLP Prefixes (For Base 2.x or later testing)	RO-Zero
p.	RsvdP_31-24 (For Base 2.x or later testing)	RO-Zero
q.	RsvdP_31-6 (For Base 1.x testing)	RO-Zero

(The MR Enable field must be written with 0, then the NumVH field must be written with the value returned by the MaxVH field, and then the MR Enable field must be written with 1 before this register is tested.)

Note: Testing this requires programming the designated Memory BAR and enabling Memory space. The value of [table] is the BAR contents (pointed to by the Port Table BIR field value) plus the offset (value in the Port Table Offset field multiplied by 8). The value of [size] is the value in the Port Table Entry Size field multiplied by 4. All values of [n] between 0 and the value in the Num Port Table Entries field minus one inclusive must be tested. The value of [offset] is the value in the PCIe Offset field multiplied by 4. If the PCIe Offset field is zero, then this register does not exist and is not tested.

Port Table Entry (n) Device Control 2 ([table] + ([n] * [size])) + [offset] + 20h — WORD

(Switch if PCIe Offset is non-zero)

- | | |
|---|------------------------|
| a. Completion Timeout Value | RO-Zero |
| b. Completion Timeout Disable | RO-Zero |
| c. ARI Forwarding Enable | RO-Zero |
| d. AtomicOp Requester Enable
(For Base 2.x or later testing) | RO-Zero |
| e. AtomicOp Egress Blocking
(For Base 2.x or later testing)
(if Port Table Entry AtomicOp Routing Supported is 1)
(if Port Table Entry AtomicOp Routing Supported is 0) | RW
RO-Zero |
| f. IDO Request Enable
(For Base 2.x or later testing) | RO-Zero |
| g. IDO Completion Enable
(For Base 2.x or later testing) | RO-Zero |
| h. LTR Mechanism Enable
(For Base 2.x or later testing)
(if Port Table Entry LTR Mechanism Supported is 1)
(if Port Table Entry LTR Mechanism Supported is 0) | RW
RO-Zero |
| i. RsvdP_12-11
(For Base 2.x or later testing) | RO-Zero |
| j. OBFF Enable
(For Base 2.x or later testing)
(if Port Table Entry OBFF Supported is 1)
(if Port Table Entry OBFF Supported is 0) | RW
RW or
RO-Zero |
| k. End-End TLP Prefix Blocking
(For Base 2.x or later testing)
(if Port Table Entry End-End TLP Prefix Supported is 1)
(if Port Table Entry End-End TLP Prefix Supported is 0) | RW
RO-Zero |
| l. RsvdP_15-6
(For Base 1.x testing) | RO-Zero |

(The MR Enable field must be written with 0, then the NumVH field must be written with the value returned by the MaxVH field, and then the MR Enable field must be written with 1 before this register is tested.)

Note: Testing this requires programming the designated Memory BAR and enabling Memory space. The value of [table] is the BAR contents (pointed to by the Port Table BIR field value) plus the offset (value in the Port Table Offset field multiplied by 8). The value of [size] is the value in the Port Table Entry Size field multiplied by 4. All values of [n] between 0 and the value in the Num Port Table Entries field minus one inclusive must be tested. The value of [offset] is the value in the PCIe Offset field multiplied by 4. If the PCIe Offset field is zero, then this register does not exist and is not tested.

Port Table Entry (n) Device Status 2 ([table] + ([n] * [size])) + [offset] + 22h — WORD

(Switch if PCIe Offset is non-zero)

- | | |
|---------------|---------|
| a. RsvdZ_15-0 | RO-Zero |
|---------------|---------|

(The MR Enable field must be written with 0, then the NumVH field must be written with the value returned by the MaxVH field, and then the MR Enable field must be written with 1 before this register is tested.)

Note: Testing this requires programming the designated Memory BAR and enabling Memory space. The value of [table] is the BAR contents (pointed to by the Port Table BIR field value) plus the offset (value in the Port Table Offset field multiplied by 8). The value of [size] is the value in the Port Table Entry Size field multiplied by 4. All values of [n] between 0 and the value in the Num Port Table Entries field minus one inclusive must be tested. The value of [offset] is the value in the PCIe Offset field multiplied by 4. If the PCIe Offset field is zero, then this register does not exist and is not tested.

Port Table Entry (n) Link Capabilities 2 ([table] + ([n] * [size])) + [offset] + 24h — DWORD

(Switch if PCIe Offset is non-zero)

- | | |
|--|---------|
| a. RsvdP_0
(For Base 3.x or later testing) | RO-Zero |
| b. Supported Link Speeds Vector
(For Base 3.x or later testing) | RO |
| c. Crosslink Supported
(For Base 3.x or later testing) | RO |
| d. RsvdP_31-9
(For Base 3.x or later testing) | RO-Zero |
| e. RsvdP_31-0
(For Base 1.x or Base 2.x testing) | RO-Zero |

(The MR Enable field must be written with 0, then the NumVH field must be written with the value returned by the MaxVH field, and then the MR Enable field must be written with 1 before this register is tested.)

Note: Testing this requires programming the designated Memory BAR and enabling Memory space. The value of [table] is the BAR contents (pointed to by the Port Table BIR field value) plus the offset (value in the Port Table Offset field multiplied by 8). The value of [size] is the value in the Port Table Entry Size field multiplied by 4. All values of [n] between 0 and the value in the Num Port Table Entries field minus one inclusive must be tested. The value of [offset] is the value in the PCIe Offset field multiplied by 4. If the PCIe Offset field is zero, then this register does not exist and is not tested.

Port Table Entry (n) Link Control 2 ([table] + ([n] * [size])) + [offset] + 28h — WORD

(Switch if PCIe Offset is non-zero)

- a. Target Link Speed
(For Base 2.x or later testing)
(if Port Table Entry Max Link Speed/Supported Link Speeds is 0001b) RWS or RO-Zero
(otherwise) RWS
Note: This test must only write supported values as reported in the Port Table Entry Max Link Speed/Supported Link Speeds field.
- b. Enter Compliance
(For Base 2.x or later testing)
(if Port Table Entry Max Link Speed/Supported Link Speeds is 0001b) RWS or RO-Zero
(if Port Table Entry Max Link Speed/Supported Link Speeds is 0010b or greater) RWS
Note: The Enter Compliance field cannot be tested for stickiness even though the register field attribute is RWS because this bit is reset to zero (upon exiting Polling.Compliance) during successful link training.
Note: Enter Compliance field cannot be tested for default value, as setting it to 1 and then doing a Hot Reset or DL_Down (link disable/enable), will cause the link to go to Compliance Pattern state and it may not return to L0 again.
- c. Hardware Autonomous Speed Disable
(For Base 2.x or later testing) RWS or RO-Zero
- d. Selectable De-emphasis
(For Base 2.x or later testing)
(if Port Direction Control is 1 (downstream port), for non-FLR testing) HwInit
(if Port Direction Control is 1 (downstream port), for FLR testing) RO
(if Port Direction Control is 0 (upstream port)) RO-Zero
- e. Transmit Margin
(For Base 2.x or later testing)
(if Port Table Entry Max Link Speed/Supported Link Speeds is 0001b) RWS or RO-Zero
(if Port Table Entry Max Link Speed/Supported Link Speeds is 0010b or greater) RWS
Note: The Transmit margin field cannot be tested for stickiness even though the register field attribute is RWS because this field is reset to zero (in Polling.Configuration) during successful link training.
- f. Enter Modified Compliance
(For Base 2.x or later testing)
(if Port Table Entry Max Link Speed/Supported Link Speeds is 0001b) RWS or RO-Zero
(if Port Table Entry Max Link Speed/Supported Link Speeds is 0010b or greater) RWS

- g. Compliance SOS
(For Base 2.x or later testing)
(if Port Table Entry Max Link Speed/Supported Link Speeds is 0001b) RWS or RO-Zero

(if Port Table Entry Max Link Speed/Supported Link Speeds is 0010b or greater) RWS
- h. Compliance Preset/De-emphasis
(For Base 3.x or later testing)
(if Port Table Entry Max Link Speed/Supported Link Speeds is 0001b) RWS or RO-Zero

(if Port Table Entry Max Link Speed/Supported Link Speeds is 0010b or greater) RWS
Note: For Base 3.x or later, this field consists of bits 15-12 of this register.
- i. Compliance De-emphasis
(For Base 2.x testing)
(if Port Table Entry Max Link Speed/Supported Link Speeds is 0001b) RWS or RO-Zero

(if Port Table Entry Max Link Speed/Supported Link Speeds is 0010b or greater) RWS
Note: For Base 2.x, this field consists of bit 12 of this register.
- j. RsvdP_15-13
(For Base 2.x testing) RO-Zero
- k. RsvdP_15-0
(For Base 1.x testing) RO-Zero

(The MR Enable field must be written with 0, then the NumVH field must be written with the value returned by the MaxVH field, and then the MR Enable field must be written with 1 before this register is tested.)

Note: Testing this requires programming the designated Memory BAR and enabling Memory space. The value of [table] is the BAR contents (pointed to by the Port Table BIR field value) plus the offset (value in the Port Table Offset field multiplied by 8). The value of [size] is the value in the Port Table Entry Size field multiplied by 4. All values of [n] between 0 and the value in the Num Port Table Entries field minus one inclusive must be tested. The value of [offset] is the value in the PCIe Offset field multiplied by 4. If the PCIe Offset field is zero, then this register does not exist and is not tested.

Port Table Entry (n) Link Status 2 ([table] + ([n] * [size])) + [offset] + 2Ah — WORD

(Switch if PCIe Offset is non-zero)

- a. Current De-emphasis Level
(For Base 2.x or later testing) RO
- b. Equalization Complete
(For Base 3.x or later testing) ROS
Note: The Equalization Complete field cannot be tested for stickiness even though the register field attribute is ROS because this bit is reset to zero on link down, and is then again updated during link equalization.

- c. Equalization Phase 1 Successful
(For Base 3.x or later testing) ROS
Note: The Equalization Phase 1 Successful field cannot be tested for stickiness even though the register field attribute is ROS because this bit is reset to zero on link down, and is then again updated during link equalization.
- d. Equalization Phase 2 Successful
(For Base 3.x or later testing) ROS
Note: The Equalization Phase 2 Successful field cannot be tested for stickiness even though the register field attribute is ROS because this bit is reset to zero on link down, and is then again updated during link equalization.
- e. Equalization Phase 3 Successful
(For Base 3.x or later testing) ROS
Note: The Equalization Phase 3 Successful field cannot be tested for stickiness even though the register field attribute is ROS because this bit is reset to zero on link down, and is then again updated during link equalization.
- f. Link Equalization Request
(For Base 3.x or later testing) RW1CS
Note: The Link Equalization Request field cannot be tested for stickiness even though the register field attribute is RW1CS because this bit is updated during link equalization.
- g. RsvdZ_15-6
(For Base 3.x or later testing) RO-Zero
- h. RsvdZ_15-1
(For Base 2.x testing) RO-Zero
- i. RsvdZ_15-0
(For Base 1.x testing) RO-Zero

(The MR Enable field must be written with 0, then the NumVH field must be written with the value returned by the MaxVH field, and then the MR Enable field must be written with 1 before this register is tested.)

Note: Testing this requires programming the designated Memory BAR and enabling Memory space. The value of [table] is the BAR contents (pointed to by the Port Table BIR field value) plus the offset (value in the Port Table Offset field multiplied by 8). The value of [size] is the value in the Port Table Entry Size field multiplied by 4. All values of [n] between 0 and the value in the Num Port Table Entries field minus one inclusive must be tested. The value of [offset] is the value in the PCIe Offset field multiplied by 4. If the PCIe Offset field is zero, then this register does not exist and is not tested.

Port Table Entry (n) Slot Capabilities 2 ([table] + ([n] * [size])) + [offset] + 2Ch — DWORD

(Switch if PCIe Offset is non-zero)

- a. RsvdP_31-0 RO-Zero

(The MR Enable field must be written with 0, then the NumVH field must be written with the value returned by the MaxVH field, and then the MR Enable field must be written with 1 before this register is tested.)

Note: Testing this requires programming the designated Memory BAR and enabling Memory space. The value of [table] is the BAR contents (pointed to by the Port Table BIR field value) plus the offset (value in the Port Table Offset field multiplied by 8). The value of [size] is the value in the Port Table Entry Size field multiplied by 4. All values of [n] between 0 and the

value in the Num Port Table Entries field minus one inclusive must be tested. The value of [offset] is the value in the PCIe Offset field multiplied by 4. If the PCIe Offset field is zero, then this register does not exist and is not tested. If Port Table Entry Slot Implemented is 0, then the entire register is tested as RO-Zero.

Port Table Entry (n) Slot Control 2 ([table] + ([n] * [size])) + [offset] + 30h — WORD

(Switch if PCIe Offset is non-zero)

a. RsvdP_15-0

RO-Zero

(The MR Enable field must be written with 0, then the NumVH field must be written with the value returned by the MaxVH field, and then the MR Enable field must be written with 1 before this register is tested.)

Note: Testing this requires programming the designated Memory BAR and enabling Memory space. The value of [table] is the BAR contents (pointed to by the Port Table BIR field value) plus the offset (value in the Port Table Offset field multiplied by 8). The value of [size] is the value in the Port Table Entry Size field multiplied by 4. All values of [n] between 0 and the value in the Num Port Table Entries field minus one inclusive must be tested. The value of [offset] is the value in the PCIe Offset field multiplied by 4. If the PCIe Offset field is zero, then this register does not exist and is not tested. If Port Table Entry Slot Implemented is 0, then the entire register is tested as RO-Zero.

Port Table Entry (n) Slot Status 2 ([table] + ([n] * [size])) + [offset] + 32h — WORD

(Switch if PCIe Offset is non-zero)

a. RsvdZ_15-0

RO-Zero

(The MR Enable field must be written with 0, then the NumVH field must be written with the value returned by the MaxVH field, and then the MR Enable field must be written with 1 before this register is tested.)

Note: Testing this requires programming the designated Memory BAR and enabling Memory space. The value of [table] is the BAR contents (pointed to by the Port Table BIR field value) plus the offset (value in the Port Table Offset field multiplied by 8). The value of [size] is the value in the Port Table Entry Size field multiplied by 4. All values of [n] between 0 and the value in the Num Port Table Entries field minus one inclusive must be tested. The value of [offset] is the value in the PCIe Offset field multiplied by 4. If the PCIe Offset field is zero, then this register does not exist and is not tested. If Port Table Entry Slot Implemented is 0, then the entire register is tested as RO-Zero.

Port Table Entry (n) Reserved ([table] + ([n] * [size])) + [offset] + 34h + ([m] * 4) — [r] DWORDS

(Switch)

(all bits)

RO-Zero

(This group of registers only exists if [r] is larger than 0, and therefore this group of registers is not tested if this condition is not true. The MR Enable field must be written with 0, then the NumVH field must be written with the value returned by the MaxVH field, and then the MR Enable field must be written with 1 before this register is tested.)

Note: Testing this requires programming the designated Memory BAR and enabling Memory space. The value of [table] is the BAR contents (pointed to by the Port Table BIR field value) plus the offset (value in the Port Table Offset field multiplied by 8). The value of [size] is the value in the Port Table Entry Size field multiplied by 4. The value of [r] is $((\text{Port Table Entry Size} * 4) - (([\text{offset}] - 1) + (13 * 4))) / 4$. All values of [n] between 0 and the value in the Num

Port Table Entries field minus one inclusive must be tested. All values of [m] between 0 and $(((((\text{Port Table Entry Size} * 4) - (([\text{offset}] - 1) + (13 * 4))) / 4) - 1)$ must be tested. The value of [offset] is the value in the PCIe Offset field multiplied by 4.

VS Interrupt Status Bitmap ([table] - 8) — 8 DWORDS

(Switch)

- | | | |
|----|--|---------|
| a. | (bits [Num VS Table Entries]-1 to 0) | |
| | (for bit [n] if VS Table Entry [n] VS Present is 1 and VS is Authorized) | RO |
| | Note: All values of [n] from 0 to [Num VS Table Entries]-1 are tested. | |
| | (for bit [n] if VS Table Entry [n] VS Present is 0) | RO-Zero |
| | Note: All values of [n] from 0 to [Num VS Table Entries]-1 are tested. | |
| | (if VS is not Authorized) | RO-Zero |
| b. | (bits (8 * 32) - 1 to [Num VS Table Entries]) | RO-Zero |

(The MR Enable field must be written with 0, then the NumVH field must be written with the value returned by the MaxVH field, and then the MR Enable field must be written with 1 before this register is tested.)

Note: Testing this requires programming the designated Memory BAR and enabling Memory space. The value of [table] is the BAR contents (pointed to by the VS Table BIR field value) plus the offset (value in the VS Table Offset field multiplied by 8).

VS Table Entry (n) VS Capability and Status ([table] + ([n] * [size])) + 00h — DWORD

(Switch)

- | | | |
|----|---|---------|
| a. | VS Present | |
| | (if VS is Authorized) | RO |
| | (if VS is not Authorized) | RO-Zero |
| b. | RsvdZ_28-1 | RO-Zero |
| c. | VS Global Key Entering Check Supported | |
| | (if VS is Authorized) | RO |
| | (if VS is not Authorized) | RO-Zero |
| d. | VS Global Key Exiting Check Supported | |
| | (if VS is Authorized) | RO |
| | (if VS is not Authorized) | RO-Zero |
| e. | VS Global Key Terminating Check Supported | |
| | (if VS is Authorized) | RO-Ones |
| | (if VS is not Authorized) | RO-Zero |

(The MR Enable field must be written with 0, then the NumVH field must be written with the value returned by the MaxVH field, and then the MR Enable field must be written with 1 before this register is tested.)

Note: Testing this requires programming the designated Memory BAR and enabling Memory space. The value of [table] is the BAR contents (pointed to by the VS Table BIR field value) plus the offset (value in the VS Table Offset field multiplied by 8). The value of [size] is the value in the VS Table Entry Size field multiplied by 4. All values of [n] between 0 and the value in the Num VS Table Entries field minus one inclusive must be tested.

VS Table Entry (n) VS Control ([table] + ([n] * [size])) + 04h — DWORD

(Switch)

- | | | |
|----|--|------------------|
| a. | VS Enable
(if VS is Authorized) | RW or
RO-Ones |
| | (if VS is not Authorized) | RO-Zero |
| b. | VS Bridge Interrupt Enable
(if VS is Authorized) | RW |
| | (if VS is not Authorized) | RO-Zero |
| c. | VS Suppress Reset Propagation
(if VS is Authorized) | RW |
| | (if VS is not Authorized) | RO-Zero |
| | Note: Changing the value of this field may reset the associated link. | |
| d. | RsvdP_15-3 | RO-Zero |
| e. | VS Global Key Value
(if VS is Authorized) | RW |
| | (if VS is not Authorized) | RO-Zero |
| f. | RsvdP_28 | RO-Zero |
| g. | VS Global Key Entering Check Enable
(if VS is Authorized and VS Global Key Entering Check Supported is 1) | RW |
| | (if VS is Authorized and VS Global Key Entering Check Supported is 0) | RO-Zero |
| | (if VS is not Authorized) | RO-Zero |
| h. | VS Global Key Exiting Check Enable
(if VS is Authorized and VS Global Key Exiting Check Supported is 1) | RW |
| | (if VS is Authorized and VS Global Key Exiting Check Supported is 0) | RO-Zero |
| | (if VS is not Authorized) | RO-Zero |
| i. | VS Global Key Terminating Check Enable
(if VS is Authorized) | RW |
| | (if VS is not Authorized) | RO-Zero |

(The MR Enable field must be written with 0, then the NumVH field must be written with the value returned by the MaxVH field, and then the MR Enable field must be written with 1 before this register is tested.)

Note: Testing this requires programming the designated Memory BAR and enabling Memory space. The value of [table] is the BAR contents (pointed to by the VS Table BIR field value) plus the offset (value in the VS Table Offset field multiplied by 8). The value of [size] is the value in the VS Table Entry Size field multiplied by 4. All values of [n] between 0 and the value in the Num VS Table Entries field minus one inclusive must be tested.

VS Table Entry (n) VS Bridge Interrupt Status ([table] + ([n] * [size])) + 08h – [r] DWORDS

(Switch)

- a. (bits [Num VS Bridge Table Entries]-1 to 0)
(for bit [n] if VS Bridge Table Entry [n] Bridge Hardware Present is 1
and VS is Authorized) RO
Note: All values of [n] from 0 to [Num VS Bridge Table Entries]-1 are tested.
(for bit [n] if VS Bridge Table Entry [n] Bridge Hardware Present is 0) RO-Zero
Note: All values of [n] from 0 to [Num VS Bridge Table Entries]-1 are tested.
(if VS is not Authorized) RO-Zero
- b. (bits ([r] * 32) - 1 to [Num VS Bridge Table Entries]) RO-Zero

(The MR Enable field must be written with 0, then the NumVH field must be written with the value returned by the MaxVH field, and then the MR Enable field must be written with 1 before this register is tested.)

Note: Testing this requires programming the designated Memory BAR and enabling Memory space. The value of [table] is the BAR contents (pointed to by the VS Table BIR field value) plus the offset (value in the VS Table Offset field multiplied by 8). The value of [size] is the value in the VS Table Entry Size field multiplied by 4. All values of [n] between 0 and the value in the Num VS Table Entries field minus one inclusive must be tested. The value of [r] is ([Num VS Bridge Table Entries] divided by 32) rounded up to the next integer number.

VS Table Entry (n) Reserved ([table] + ([n] * [size])) + 08h + [offset] + ([m] * 4) – [r] DWORDS

(Switch)

(all bits) RO-Zero

(This group of registers only exists if [r] is larger than 0, and therefore this group of registers is not tested if this condition is not true. The MR Enable field must be written with 0, then the NumVH field must be written with the value returned by the MaxVH field, and then the MR Enable field must be written with 1 before this register is tested.)

Note: Testing this requires programming the designated Memory BAR and enabling Memory space. The value of [table] is the BAR contents (pointed to by the VS Table BIR field value) plus the offset (value in the VS Table Offset field multiplied by 8). The value of [size] is the value in the VS Table Entry Size field multiplied by 4. The value of [r] is $((([VS\ Table\ Entry\ Size] * 4) - ([offset] + (2 * 4))) / 4)$. All values of [n] between 0 and the value in the Num VS Table Entries field minus one inclusive must be tested. All values of [m] between 0 and $((([VS\ Table\ Entry\ Size] * 4) - ([offset] + (2 * 4))) / 4) - 1$ must be tested. The value of [offset] is ([Num VS Bridge Table Entries] divided by 32) rounded up to the next integer number multiplied by 4.

VS Bridge Table Entry (n, x) VS Bridge Capability and Status ([table] + (([n] * [bsize]) + ([x] * [size]))) + 00h — DWORD

(Switch)

a. Bridge Hardware Present (if VS is Authorized) (if VS is not Authorized)	RO RO-Zero
b. Hot-Plug Hardware Present (if VS is Authorized) (if VS is not Authorized)	RO RO-Zero
c. RsvdZ_3-2	RO-Zero
d. Max Payload Size Supported (if VS is Authorized) (if VS is not Authorized)	RO RO-Zero
e. RsvdZ_7	RO-Zero
f. Num VC Resources Hardware Present (if VS is Authorized) (if VS is not Authorized)	RO RO-Zero
g. RsvdZ_15-11	RO-Zero
h. Link in Reset (if VS is Authorized and Port Mapped to Bridge is 1) (if VS is Authorized and Port Mapped to Bridge is 0) (if VS is not Authorized)	RO RO-Ones RO-Zero
i. PME Turn Off State (if VS is Authorized) (if VS is not Authorized)	RO RO-Zero
j. RsvdZ_26-18	RO-Zero
k. PME Turn Off State Changed (if VS is Authorized) (if VS is not Authorized)	RW1C RO-Zero
l. Power Controller State Changed (if VS is Authorized) (if VS is not Authorized)	RW1C RO-Zero
m. Power Indicator State Changed (if VS is Authorized) (if VS is not Authorized)	RW1C RO-Zero
n. Attention Indicator State Changed (if VS is Authorized) (if VS is not Authorized)	RW1C RO-Zero
o. VC Config Changed (if VS is Authorized) (if VS is not Authorized)	RW1C RO-Zero

(The MR Enable field must be written with 0, then the NumVH field must be written with the value returned by the MaxVH field, and then the MR Enable field must be written with 1 before this register is tested.)

Note: Testing this requires programming the designated Memory BAR and enabling Memory space. The value of [table] is the BAR contents (pointed to by the VS Bridge Table BIR field value) plus the offset (value in the VS Bridge Table Offset field multiplied by 8). The value of

[bsize] is the value in the Num VS Bridge Table Entries field multiplied by the VS Bridge Table Entry Size field multiplied by 4. The value of [size] is the value in the VS Bridge Table Entry Size field multiplied by 4. All values of [x] between 0 and the value in the Num VS Bridge Table Entries field minus one inclusive must be tested. All values of [n] between 0 and the value in the Num VS Table Entries field minus one inclusive must be tested.

VS Bridge Table Entry (n, x) VS Bridge Control 1 ([table] + (([n] * [bsize]) + ([x] * [size]))) + 04h — DWORD

(Switch)

- | | |
|--|---------|
| a. Bridge Enable | |
| (if VS is Authorized and Bridge Hardware Present is 1) | RW |
| (if VS is Authorized and Bridge Hardware Present is 0) | RO |
| (if VS is not Authorized) | RO-Zero |
| b. Bridge Controls Physical Link | |
| (if VS is Authorized) | RW or |
| | RO-Zero |
| (if VS is not Authorized) | RO-Zero |
| c. VC Config Changed Interrupt Enable | |
| (if VS is Authorized) | RW |
| (if VS is not Authorized) | RO-Zero |
| d. PME Turn Off State Change Interrupt Enable | |
| (if VS is Authorized) | RW |
| (if VS is not Authorized) | RO-Zero |
| e. RsvdP_14-4 | RO-Zero |
| f. Port Mapped to Bridge | |
| (if VS is Authorized and Bridge Enable is 1) | RW |
| (if VS is Authorized and Bridge Enable is 0) | RO-Zero |
| (if VS is not Authorized) | RO-Zero |
| g. Bridge VHN | |
| (if VS is Authorized) | |
| after first writing Port Mapped to Bridge with 0 | RW |
| after first writing Port Mapped to Bridge with 1 | RO |
| (if VS is not Authorized) | RO-Zero |
| h. Bridge Port | |
| (if VS is Authorized) | |
| after first writing Port Mapped to Bridge with 0 | RW |
| after first writing Port Mapped to Bridge with 1 | RO |
| (if VS is not Authorized) | RO-Zero |

(The MR Enable field must be written with 0, then the NumVH field must be written with the value returned by the MaxVH field, and then the MR Enable field must be written with 1 before this register is tested.)

Note: Testing this requires programming the designated Memory BAR and enabling Memory space. The value of [table] is the BAR contents (pointed to by the VS Bridge Table BIR field value) plus the offset (value in the VS Bridge Table Offset field multiplied by 8). The value of [bsize] is the value in the Num VS Bridge Table Entries field multiplied by the VS Bridge Table Entry Size field multiplied by 4. The value of [size] is the value in the VS Bridge Table Entry Size field multiplied by 4. All values of [x] between 0 and the value in the Num VS Bridge Table

Entries field minus one inclusive must be tested. All values of [n] between 0 and the value in the Num VS Table Entries field minus one inclusive must be tested.

VS Bridge Table Entry (n, x) VS Bridge Control 2 ([table] + (([n] * [bsize]) + ([x] * [size]))) + 08h — DWORD

(Switch)

- | | |
|--|---------|
| a. RsvdP_3-0 | RO-Zero |
| b. Max Payload Size Offered
(if VS is Authorized) | RW |
| (if VS is not Authorized) | RO-Zero |
| c. RsvdP_7 | RO-Zero |
| d. Extended VC Count
(if VS is Authorized) | RW |
| (if VS is not Authorized) | RO-Zero |
| e. RsvdP_11 | RO-Zero |
| f. Low Priority Extended VC Count
(if VS is Authorized) | RW |
| (if VS is not Authorized) | RO-Zero |
| g. RsvdP_31-15 | RO-Zero |

(The MR Enable field must be written with 0, then the NumVH field must be written with the value returned by the MaxVH field, and then the MR Enable field must be written with 1 before this register is tested.)

Note: Testing this requires programming the designated Memory BAR and enabling Memory space. The value of [table] is the BAR contents (pointed to by the VS Bridge Table BIR field value) plus the offset (value in the VS Bridge Table Offset field multiplied by 8). The value of [bsize] is the value in the Num VS Bridge Table Entries field multiplied by the VS Bridge Table Entry Size field multiplied by 4. The value of [size] is the value in the VS Bridge Table Entry Size field multiplied by 4. All values of [x] between 0 and the value in the Num VS Bridge Table Entries field minus one inclusive must be tested. All values of [n] between 0 and the value in the Num VS Table Entries field minus one inclusive must be tested.

VS Bridge Table Entry (n, x) Virtual Hot-Plug Signals Interface 1 ([table] + (([n] * [bsize]) + ([x] * [size]))) + 0Ch — DWORD

(Switch)

- | | |
|---|---------|
| a. RsvdP_0 | RO-Zero |
| b. Virtual Power Controller Present
(if VS is Authorized and Hot-Plug Hardware Present is 1) | RW |
| (if VS is not Authorized) | RO-Zero |
| c. RsvdP_4-2 | RO-Zero |
| d. Virtual Hot Plug Surprise
(if VS is Authorized and Hot-Plug Hardware Present is 1) | RW |
| (if VS is not Authorized) | RO-Zero |
| e. Virtual Hot Plug Capable
(if VS is Authorized and Hot-Plug Hardware Present is 1) | RW |
| (if VS is not Authorized) | RO-Zero |
| f. RsvdP_18-7 | RO-Zero |

- g. Virtual Slot Number
 (if VS is Authorized and Hot-Plug Hardware Present is 1) RW
 (if VS is not Authorized) RO-Zero

(The MR Enable field must be written with 0, then the NumVH field must be written with the value returned by the MaxVH field, and then the MR Enable field must be written with 1 before this register is tested.)

Note: Testing this requires programming the designated Memory BAR and enabling Memory space. The value of [table] is the BAR contents (pointed to by the VS Bridge Table BIR field value) plus the offset (value in the VS Bridge Table Offset field multiplied by 8). The value of [bsize] is the value in the Num VS Bridge Table Entries field multiplied by the VS Bridge Table Entry Size field multiplied by 4. The value of [size] is the value in the VS Bridge Table Entry Size field multiplied by 4. All values of [x] between 0 and the value in the Num VS Bridge Table Entries field minus one inclusive must be tested. All values of [n] between 0 and the value in the Num VS Table Entries field minus one inclusive must be tested.

VS Bridge Table Entry (n, x) Virtual Hot-Plug Signals Interface 2 ([table] + (([n] * [bsize]) + ([x] * [size]))) + 10h — DWORD
 (Switch)

- a. Virtual Power Indicator State
 (if VS is Authorized and Hot-Plug Hardware Present is 1) RO
 (if VS is not Authorized) RO-Zero
- b. Virtual Attention Indicator State
 (if VS is Authorized and Hot-Plug Hardware Present is 1) RO
 (if VS is not Authorized) RO-Zero
- c. Virtual Power Controller State
 (if VS is Authorized and Hot-Plug Hardware Present is 1) RO
 (if VS is not Authorized) RO-Zero
- d. RsvdP_7-5 RO-Zero
- e. Virtual Slot Implemented
 (if VS is Authorized and Hot-Plug Hardware Present is 1) RW
 (if VS is Authorized and Hot-Plug Hardware Present is 0) RO-Zero
 (if VS is not Authorized) RO-Zero
- f. Hot Plug Signals Interrupt Enable
 (if VS is Authorized and Hot-Plug Hardware Present is 1) RW
 (if VS is Authorized and Hot-Plug Hardware Present is 0) RO-Zero
 (if VS is not Authorized) RO-Zero
- g. RsvdP_15-10 RO-Zero
- h. Virtual Discard Ingress Request
 (if VS is Authorized) RW
 (if VS is not Authorized) RO-Zero
- i. Virtual Presence Detect State
 (if VS is Authorized and Hot-Plug Hardware Present is 1) RW
 (if VS is not Authorized) RO-Zero
- j. Virtual Force Reset
 (if VS is Authorized and Port Direction Control is 1 (downstream port)) RW
 (if VS is Authorized and Port Direction Control is 0 (upstream port)) RO-Zero
 (if VS is not Authorized) RO-Zero

- | | |
|----------------------------------|---------|
| k. Push Virtual Attention Button | RO-Zero |
| l. Signal Virtual Power Fault | RO-Zero |
| m. RsvdP_31-21 | RO-Zero |

(The MR Enable field must be written with 0, then the NumVH field must be written with the value returned by the MaxVH field, and then the MR Enable field must be written with 1 before this register is tested.)

Note: Testing this requires programming the designated Memory BAR and enabling Memory space. The value of [table] is the BAR contents (pointed to by the VS Bridge Table BIR field value) plus the offset (value in the VS Bridge Table Offset field multiplied by 8). The value of [bsize] is the value in the Num VS Bridge Table Entries field multiplied by the VS Bridge Table Entry Size field multiplied by 4. The value of [size] is the value in the VS Bridge Table Entry Size field multiplied by 4. All values of [x] between 0 and the value in the Num VS Bridge Table Entries field minus one inclusive must be tested. All values of [n] between 0 and the value in the Num VS Table Entries field minus one inclusive must be tested.

VS Bridge Table Entry (n, x) VS Bridge VC ID to VL Map 1 ([table] + (([n] * [bsize]) + ([x] * [size]))) + 14h — DWORD

(Switch)

- | | |
|--|---------|
| a. VC0 VL Map | |
| (if VS is Authorized and MaxVL is non-zero) | |
| after first writing VC0 VL Map Enable with 1 | RW |
| after first writing VC0 VL Map Enable with 0 | RO |
| (if VS is Authorized and MaxVL is 0) | RO-Zero |
| (if VS is not Authorized) | RO-Zero |
| b. RsvdP_3 | RO-Zero |
| c. VC0 VL Map Enable | |
| (if VS is Authorized and MaxVL is non-zero) | RW |
| (if VS is Authorized and MaxVL is 0) | RO-Zero |
| (if VS is not Authorized) | RO-Zero |
| d. RsvdP_7-5 | RO-Zero |
| e. VC1 VL Map | |
| (if VS is Authorized and MaxVL is non-zero | |
| and Num VC Resources Hardware Present is greater than or equal to 1) | |
| after first writing VC1 VL Map Enable with 1 | RW |
| after first writing VC1 VL Map Enable with 0 | RO |
| (if VS is Authorized and MaxVL is 0) | RO-Zero |
| (if VS is Authorized | |
| and Num VC Resources Hardware Present is less than 1) | RO-Zero |
| (if VS is not Authorized) | RO-Zero |
| f. RsvdP_11 | RO-Zero |
| g. VC1 VL Map Enable | |
| (if VS is Authorized and MaxVL is non-zero | |
| and Num VC Resources Hardware Present is greater than or equal to 1) | RW |
| (if VS is Authorized and MaxVL is 0) | RO-Zero |
| (if VS is Authorized | |
| and Num VC Resources Hardware Present is less than 1) | RO-Zero |
| (if VS is not Authorized) | RO-Zero |

h.	RsvdP_15-13	RO-Zero
i.	VC2 VL Map	
	(if VS is Authorized and MaxVL is non-zero	
	and Num VC Resources Hardware Present is greater than or equal to 2)	
	after first writing VC2 VL Map Enable with 1	RW
	after first writing VC2 VL Map Enable with 0	RO
	(if VS is Authorized and MaxVL is 0)	RO-Zero
	(if VS is Authorized	
	and Num VC Resources Hardware Present is less than 2)	RO-Zero
	(if VS is not Authorized)	RO-Zero
j.	RsvdP_19	RO-Zero
k.	VC2 VL Map Enable	
	(if VS is Authorized and MaxVL is non-zero	
	and Num VC Resources Hardware Present is greater than or equal to 2)	RW
	(if VS is Authorized and MaxVL is 0)	RO-Zero
	(if VS is Authorized	
	and Num VC Resources Hardware Present is less than 2)	RO-Zero
	(if VS is not Authorized)	RO-Zero
l.	RsvdP_23-21	RO-Zero
m.	VC3 VL Map	
	(if VS is Authorized and MaxVL is non-zero	
	and Num VC Resources Hardware Present is greater than or equal to 3)	
	after first writing VC3 VL Map Enable with 1	RW
	after first writing VC3 VL Map Enable with 0	RO
	(if VS is Authorized and MaxVL is 0)	RO-Zero
	(if VS is Authorized	
	and Num VC Resources Hardware Present is less than 3)	RO-Zero
	(if VS is not Authorized)	RO-Zero
n.	RsvdP_27	RO-Zero
o.	VC3 VL Map Enable	
	(if VS is Authorized and MaxVL is non-zero	
	and Num VC Resources Hardware Present is greater than or equal to 3)	RW
	(if VS is Authorized and MaxVL is 0)	RO-Zero
	(if VS is Authorized	
	and Num VC Resources Hardware Present is less than 3)	RO-Zero
	(if VS is not Authorized)	RO-Zero
p.	RsvdP_31-29	RO-Zero

(The MR Enable field must be written with 0, then the NumVH field must be written with the value returned by the MaxVH field, and then the MR Enable field must be written with 1 before this register is tested.)

Note: Testing this requires programming the designated Memory BAR and enabling Memory space. The value of [table] is the BAR contents (pointed to by the VS Bridge Table BIR field value) plus the offset (value in the VS Bridge Table Offset field multiplied by 8). The value of [bsize] is the value in the Num VS Bridge Table Entries field multiplied by the VS Bridge Table Entry Size field multiplied by 4. The value of [size] is the value in the VS Bridge Table Entry Size field multiplied by 4. All values of [x] between 0 and the value in the Num VS Bridge Table

Entries field minus one inclusive must be tested. All values of [n] between 0 and the value in the Num VS Table Entries field minus one inclusive must be tested.

VS Bridge Table Entry (n, x) VS Bridge VC ID to VL Map 2 ([table] + (([n] * [bsize]) + ([x] * [size]))) + 18h — DWORD

(Switch)

- | | |
|--|---------|
| a. VC4 VL Map | |
| (if VS is Authorized and MaxVL is non-zero | |
| and Num VC Resources Hardware Present is greater than or equal to 4) | |
| after first writing VC4 VL Map Enable with 1 | RW |
| after first writing VC4 VL Map Enable with 0 | RO |
| (if VS is Authorized and MaxVL is 0) | RO-Zero |
| (if VS is Authorized | |
| and Num VC Resources Hardware Present is less than 4) | RO-Zero |
| (if VS is not Authorized) | RO-Zero |
| b. RsvdP_3 | RO-Zero |
| c. VC4 VL Map Enable | |
| (if VS is Authorized and MaxVL is non-zero | |
| and Num VC Resources Hardware Present is greater than or equal to 4) | |
| (if VS is Authorized and MaxVL is 0) | RW |
| (if VS is Authorized | RO-Zero |
| and Num VC Resources Hardware Present is less than 4) | |
| (if VS is not Authorized) | RO-Zero |
| d. RsvdP_7-5 | RO-Zero |
| e. VC5 VL Map | |
| (if VS is Authorized and MaxVL is non-zero | |
| and Num VC Resources Hardware Present is greater than or equal to 5) | |
| after first writing VC5 VL Map Enable with 1 | RW |
| after first writing VC5 VL Map Enable with 0 | RO |
| (if VS is Authorized and MaxVL is 0) | RO-Zero |
| (if VS is Authorized | |
| and Num VC Resources Hardware Present is less than 5) | RO-Zero |
| (if VS is not Authorized) | RO-Zero |
| f. RsvdP_11 | RO-Zero |
| g. VC5 VL Map Enable | |
| (if VS is Authorized and MaxVL is non-zero | |
| and Num VC Resources Hardware Present is greater than or equal to 5) | |
| (if VS is Authorized and MaxVL is 0) | RW |
| (if VS is Authorized | RO-Zero |
| and Num VC Resources Hardware Present is less than 5) | |
| (if VS is not Authorized) | RO-Zero |
| h. RsvdP_15-13 | RO-Zero |

- i. VC6 VL Map
 - (if VS is Authorized and MaxVL is non-zero
and Num VC Resources Hardware Present is greater than or equal to 6)
after first writing VC6 VL Map Enable with 1 RW
 - after first writing VC6 VL Map Enable with 0 RO
 - (if VS is Authorized and MaxVL is 0) RO-Zero
 - (if VS is Authorized
and Num VC Resources Hardware Present is less than 6) RO-Zero
 - (if VS is not Authorized) RO-Zero
- j. RsvdP_19 RO-Zero
- k. VC6 VL Map Enable
 - (if VS is Authorized and MaxVL is non-zero
and Num VC Resources Hardware Present is greater than or equal to 6) RW
 - (if VS is Authorized and MaxVL is 0) RO-Zero
 - (if VS is Authorized
and Num VC Resources Hardware Present is less than 6) RO-Zero
 - (if VS is not Authorized) RO-Zero
- l. RsvdP_23-21 RO-Zero
- m. VC7 VL Map
 - (if VS is Authorized and MaxVL is non-zero
and Num VC Resources Hardware Present is equal to 7)
after first writing VC7 VL Map Enable with 1 RW
 - after first writing VC7 VL Map Enable with 0 RO
 - (if VS is Authorized and MaxVL is 0) RO-Zero
 - (if VS is Authorized
and Num VC Resources Hardware Present is less than 7) RO-Zero
 - (if VS is not Authorized) RO-Zero
- n. RsvdP_27 RO-Zero
- o. VC7 VL Map Enable
 - (if VS is Authorized and MaxVL is non-zero
and Num VC Resources Hardware Present is equal to 7) RW
 - (if VS is Authorized and MaxVL is 0) RO-Zero
 - (if VS is Authorized
and Num VC Resources Hardware Present is less than 7) RO-Zero
 - (if VS is not Authorized) RO-Zero
- p. RsvdP_31-29 RO-Zero

(The MR Enable field must be written with 0, then the NumVH field must be written with the value returned by the MaxVH field, and then the MR Enable field must be written with 1 before this register is tested.)

Note: Testing this requires programming the designated Memory BAR and enabling Memory space. The value of [table] is the BAR contents (pointed to by the VS Bridge Table BIR field value) plus the offset (value in the VS Bridge Table Offset field multiplied by 8). The value of [bsize] is the value in the Num VS Bridge Table Entries field multiplied by the VS Bridge Table Entry Size field multiplied by 4. The value of [size] is the value in the VS Bridge Table Entry Size field multiplied by 4. All values of [x] between 0 and the value in the Num VS Bridge Table Entries field minus one inclusive must be tested. All values of [n] between 0 and the value in the Num VS Table Entries field minus one inclusive must be tested.

VS Bridge Table Table Entry (n, x) Reserved ([table] + (([n] * [bsize]) + ([x] * [size]))) + 1Ch — 4 DWORDS

(Switch)

- | | |
|-------------------------------|---------|
| a. Reserved register (1st DW) | RO-Zero |
| b. Reserved register (2nd DW) | RO-Zero |
| c. Reserved register (3rd DW) | RO-Zero |
| d. Reserved register (4th DW) | RO-Zero |

(The MR Enable field must be written with 0, then the NumVH field must be written with the value returned by the MaxVH field, and then the MR Enable field must be written with 1 before this register is tested.)

Note: Testing this requires programming the designated Memory BAR and enabling Memory space. The value of [table] is the BAR contents (pointed to by the VS Bridge Table BIR field value) plus the offset (value in the VS Bridge Table Offset field multiplied by 8). The value of [bsize] is the value in the Num VS Bridge Table Entries field multiplied by the VS Bridge Table Entry Size field multiplied by 4. The value of [size] is the value in the VS Bridge Table Entry Size field multiplied by 4. All values of [x] between 0 and the value in the Num VS Bridge Table Entries field minus one inclusive must be tested. All values of [n] between 0 and the value in the Num VS Table Entries field minus one inclusive must be tested.

VS Bridge Table Table Entry (n, x) VC Resource State (m) ([table] + (([n] * [bsize]) + ([x] * [size]))) + 2Ch + ([m] * 4) — DWORD

(Switch)

- | | |
|--|---------|
| a. VC Resource [m] VC Enabled | |
| (if [m] is 0) | RO-Ones |
| (if [m] is non-zero | |
| and [m] is less than or equal to Num VC Resources Hardware Present | |
| and [m] is less than or equal to Extended VC Count) | RO |
| (if [m] is nonzero | |
| and [m] is less than or equal to Num VC Resources Hardware Present | |
| and [m] is greater than Extended VC Count) | RO-Zero |
| b. VC Resource [m] VC Negotiation Pending | |
| (if [m] is less than or equal to Num VC Resources Hardware Present | |
| and [m] is less than or equal to Extended VC Count) | RO |
| (if [m] is less than or equal to Num VC Resources Hardware Present | |
| and [m] is greater than Extended VC Count) | RO-Zero |
| c. RsvdZ_3-2 | RO-Zero |
| d. VC Resource [m] VC ID | |
| (if [m] is 0) | RO-Zero |
| (if [m] is non-zero | |
| and [m] is less than or equal to Num VC Resources Hardware Present | |
| and [m] is less than or equal to Extended VC Count) | RO |
| (if [m] is nonzero | |
| and [m] is less than or equal to Num VC Resources Hardware Present | |
| and [m] is greater than Extended VC Count) | RO-Zero |

- | | |
|--|---------|
| e. RsvdZ_15-7 | RO-Zero |
| f. VC Resource [m] TC to VC Map | |
| (if [m] is 0) | RO |
| (if [m] is non-zero | |
| and [m] is less than or equal to Num VC Resources Hardware Present | |
| and [m] is less than or equal to Extended VC Count) | RO |
| (if [m] is nonzero | |
| and [m] is less than or equal to Num VC Resources Hardware Present | |
| and [m] is greater than Extended VC Count) | RO-Zero |
| g. RsvdZ_31-24 | RO-Zero |

(The MR Enable field must be written with 0, then the NumVH field must be written with the value returned by the MaxVH field, and then the MR Enable field must be written with 1 before this register is tested. The register group is tested twice if the Num VC Resources Hardware Present field is greater than 0. The first time it is tested after the Extended VC Count field is written with the value returned in the Num VC Resources Hardware Present field. The second time it is tested after the Extended VC Count field is written with 000b, but only if the Num VC Resources Hardware Present field is greater than 0.)

Note: Testing this requires programming the designated Memory BAR and enabling Memory space. The value of [table] is the BAR contents (pointed to by the VS Bridge Table BIR field value) plus the offset (value in the VS Bridge Table Offset field multiplied by 8). The value of [bsize] is the value in the Num VS Bridge Table Entries field multiplied by the VS Bridge Table Entry Size field multiplied by 4. The value of [size] is the value in the VS Bridge Table Entry Size field multiplied by 4. All values of [x] between 0 and the value in the Num VS Bridge Table Entries field minus one inclusive must be tested. All values of [n] between 0 and the value in the Num VS Table Entries field minus one inclusive must be tested. All values of [m] between 0 and the value in the Num VC Resources Hardware Present field inclusive must be tested.

Statistics Descriptor Table Entry (n) Statistics Descriptor Table Entry [table] + 00h – 8 DWORDS

(Endpoint and Switch)

- | | |
|--|---------|
| a. Standard S-Bits (bits 127-0) | RO |
| b. Group 1 Vendor Specific S-Bits (bits 167-128) | RO |
| c. Group 1 Collection ID (bits 175-168) | |
| (if any Group 1 Vendor Specific S-Bits are non-zero) | RO |
| (if all Group 1 Vendor Specific S-Bits are zero) | RO-Zero |
| d. Group 1 Vendor ID (bits 191-176) | |
| (if any Group 1 Vendor Specific S-Bits are non-zero) | RO |
| (if all Group 1 Vendor Specific S-Bits are zero) | RO-Zero |
| e. Group 2 Vendor Specific S-Bits (bits 231-192) | RO |
| f. Group 2 Collection ID (bits 239-232) | |
| (if any Group 2 Vendor Specific S-Bits are non-zero) | RO |
| (if all Group 2 Vendor Specific S-Bits are zero) | RO-Zero |
| g. Group 2 Vendor ID (bits 255-240) | |
| (if any Group 2 Vendor Specific S-Bits are non-zero) | RO |
| (if all Group 2 Vendor Specific S-Bits are zero) | RO-Zero |

(This group of registers only exists if the Number of Statistics Descriptors field is non-zero, and therefore this group of registers is not tested if this condition is not true. The MR Enable field

must be written with 0, then the NumVH field must be written with the value returned by the MaxVH field, and then the MR Enable field must be written with 1 before this register is tested.) Note: Testing this requires programming the designated Memory BAR and enabling Memory space. The value of [table] is the BAR contents (pointed to by the Statistics Descriptor Table BIR field value) plus the offset (value in the Statistics Descriptor Table Offset field multiplied by 8). All values of [n] between 0 and the value in the Number of Statistics Descriptors field minus one inclusive must be tested.

Statistics Block Table Entry (n) Statistics Block Capability [table] + 00h –DWORD
(Endpoint and Switch)

- | | |
|----------------------------------|---------|
| a. Statistics Block Status | RO |
| b. RsvdZ_15-2 | RO-Zero |
| c. Statistics Counter Table Size | RO |

(This group of registers only exists if the Number of Statistics Blocks field is non-zero, and therefore this group of registers is not tested if this condition is not true. The MR Enable field must be written with 0, then the NumVH field must be written with the value returned by the MaxVH field, and then the MR Enable field must be written with 1 before this register is tested.) Note: Testing this requires programming the designated Memory BAR and enabling Memory space. The value of [table] is the BAR contents (pointed to by the Statistics Block Table BIR field value) plus the offset (value in the Statistics Block Table Offset field multiplied by 8). All values of [n] between 0 and the value in the Number of Statistics Blocks field minus one inclusive must be tested.

Statistics Block Table Entry (n) Statistics Counter Table Offset [table] + 04h –DWORD
(Endpoint and Switch)

- | | |
|------------------------------------|---------|
| a. RsvdZ_3-0 | RO-Zero |
| b. Statistics Counter Table Offset | RO |

(This group of registers only exists if the Number of Statistics Blocks field is non-zero, and therefore this group of registers is not tested if this condition is not true. The MR Enable field must be written with 0, then the NumVH field must be written with the value returned by the MaxVH field, and then the MR Enable field must be written with 1 before this register is tested.) Note: Testing this requires programming the designated Memory BAR and enabling Memory space. The value of [table] is the BAR contents (pointed to by the Statistics Block Table BIR field value) plus the offset (value in the Statistics Block Table Offset field multiplied by 8). All values of [n] between 0 and the value in the Number of Statistics Blocks field minus one inclusive must be tested.

Statistics Block Table Entry (n) Statistics Waiting Period [table] + 08h –DWORD
(Endpoint and Switch)

- | | |
|------------------------------|---------|
| a. Statistics Waiting Period | RW |
| b. RsvdP_31-16 | RO-Zero |

(This group of registers only exists if the Number of Statistics Blocks field is non-zero, and therefore this group of registers is not tested if this condition is not true. The MR Enable field must be written with 0, then the NumVH field must be written with the value returned by the MaxVH field, and then the MR Enable field must be written with 1 before this register is tested.) Note: Testing this requires programming the designated Memory BAR and enabling Memory space. The value of [table] is the BAR contents (pointed to by the Statistics Block Table BIR field value) plus the offset (value in the Statistics Block Table Offset field multiplied by 8). All

values of [n] between 0 and the value in the Number of Statistics Blocks field minus one inclusive must be tested.

Statistics Block Table Entry (n) Statistics Counting Period [table] + 0Ch –DWORD
(Endpoint and Switch)

- | | |
|-------------------------------|---------|
| a. Statistics Counting Period | RW |
| b. RsvdP_31-24 | RO-Zero |

(This group of registers only exists if the Number of Statistics Blocks field is non-zero, and therefore this group of registers is not tested if this condition is not true. The MR Enable field must be written with 0, then the NumVH field must be written with the value returned by the MaxVH field, and then the MR Enable field must be written with 1 before this register is tested.)
Note: Testing this requires programming the designated Memory BAR and enabling Memory space. The value of [table] is the BAR contents (pointed to by the Statistics Block Table BIR field value) plus the offset (value in the Statistics Block Table Offset field multiplied by 8). All values of [n] between 0 and the value in the Number of Statistics Blocks field minus one inclusive must be tested.

Statistics Block Table Entry (x) Statistics Counter Table Entry (n) Statistics Capability and Control ([table, x] + ([n] * [size])) + 00h – DWORD
(Endpoint and Switch)

- | | |
|---|------------------|
| a. Port Number
(for Switches)
(for Endpoints) | RO
RO-Zero |
| b. Statistics Descriptor Index | RO |
| c. Counter Width | |
| d. RsvdP_22 | RO-Zero |
| e. Counter Enable | RW or
RO-Ones |
| f. Statistics Select | RW or RO |

(This group of registers only exists if the Number of Statistics Blocks field is non-zero, and therefore this group of registers is not tested if this condition is not true. The MR Enable field must be written with 0, then the NumVH field must be written with the value returned by the MaxVH field, and then the MR Enable field must be written with 1 before this register is tested.)
Note: Testing this requires programming the designated Memory BAR and enabling Memory space. The value of [table, x] is the BAR contents (pointed to by the Statistics Block Table BIR field value) plus the offset (for Statistics Block Table Entry (x) the value in the Statistics Counter Table Offset field multiplied by 16). The value of [size] is for Statistics Block Table Entry (x) the value in the Statistics Counter Table Size field multiplied by 16. All values of [x] between 0 and the value in the Number of Statistics Blocks field minus one inclusive must be tested. All values of [n] between 0 and the value in the Statistics Counter Table Size field minus one inclusive must be tested.

Statistics Block Table Entry (x) Statistics Counter Table Entry (n) Statistics Filter Enable and Control ([table, x] + ([n] * [size])) + 04h – DWORD

(Endpoint and Switch)

- a. Filter Enable and Control

RW

(This group of registers only exists if the Number of Statistics Blocks field is non-zero, and therefore this group of registers is not tested if this condition is not true. The MR Enable field must be written with 0, then the NumVH field must be written with the value returned by the MaxVH field, and then the MR Enable field must be written with 1 before this register is tested.)

Note: Testing this requires programming the designated Memory BAR and enabling Memory space. The value of [table, x] is the BAR contents (pointed to by the Statistics Block Table BIR field value) plus the offset (for Statistics Block Table Entry (x) the value in the Statistics Counter Table Offset field multiplied by 16). The value of [size] is for Statistics Block Table Entry (x) the value in the Statistics Counter Table Size field multiplied by 16. All values of [x] between 0 and the value in the Number of Statistics Blocks field minus one inclusive must be tested. All values of [n] between 0 and the value in the Statistics Counter Table Size field minus one inclusive must be tested.

Statistics Block Table Entry (x) Statistics Counter Table Entry (n) Statistics Counter Low ([table, x] + ([n] * [size])) + 08h – DWORD

(Endpoint and Switch)

- | | |
|---|---------|
| a. (if Counter Width is greater than 31: bits 31 to 0) | RO |
| b. (if Counter Width is less than 32: bits [Counter Width] to 0) | RO |
| c. (if Counter Width is less than 31: bits 31 to [Counter Width] + 1) | RO-Zero |

(This group of registers only exists if the Number of Statistics Blocks field is non-zero, and therefore this group of registers is not tested if this condition is not true. The MR Enable field must be written with 0, then the NumVH field must be written with the value returned by the MaxVH field, and then the MR Enable field must be written with 1 before this register is tested.)

Note: Testing this requires programming the designated Memory BAR and enabling Memory space. The value of [table, x] is the BAR contents (pointed to by the Statistics Block Table BIR field value) plus the offset (for Statistics Block Table Entry (x) the value in the Statistics Counter Table Offset field multiplied by 16). The value of [size] is for Statistics Block Table Entry (x) the value in the Statistics Counter Table Size field multiplied by 16. All values of [x] between 0 and the value in the Number of Statistics Blocks field minus one inclusive must be tested. All values of [n] between 0 and the value in the Statistics Counter Table Size field minus one inclusive must be tested.

Statistics Block Table Entry (x) Statistics Counter Table Entry (n) Statistics Counter High ([table, x] + ([n] * [size])) + 0Ch – DWORD

(Endpoint and Switch)

- | | |
|--|---------|
| a. (if Counter Width is 63: bits 31 to 0) | RO |
| b. (if Counter Width is less than 63 and greater than 31: bits [Counter Width]-32 to 0) | RO |
| c. (if Counter Width is less than 63 and greater than 31: (bits 31 to [Counter Width]-32 + 1)) | RO-Zero |
| d. (if Counter Width is less than 32: bits 31 to 0) | RO-Zero |

(This group of registers only exists if the Number of Statistics Blocks field is non-zero, and therefore this group of registers is not tested if this condition is not true. The MR Enable field must be written with 0, then the NumVH field must be written with the value returned by the MaxVH field, and then the MR Enable field must be written with 1 before this register is tested.)

Note: Testing this requires programming the designated Memory BAR and enabling Memory space. The value of [table, x] is the BAR contents (pointed to by the Statistics Block Table BIR field value) plus the offset (for Statistics Block Table Entry (x) the value in the Statistics Counter Table Offset field multiplied by 16). The value of [size] is for Statistics Block Table Entry (x) the value in the Statistics Counter Table Size field multiplied by 16. All values of [x] between 0 and the value in the Number of Statistics Blocks field minus one inclusive must be tested. All values of [n] between 0 and the value in the Statistics Counter Table Size field minus one inclusive must be tested.

11. The following default value checks are performed:

MR-IOV Control Register Default Value (Offset 08h) — DWORD

(Endpoint)

- | | |
|---|-----|
| a. Function Table Interrupt Enable | 0 |
| b. Statistics Interrupt Enable | 0 |
| c. MR Uncorrectable Fatal TLP Error Interrupt Enable | 0 |
| d. MR Uncorrectable Global Key Error Interrupt Enable | 0 |
| e. MR Enable | 0 |
| f. VL Enable | |
| (if Is Main BF is 1 and MR Enable is written with 1) | 01h |
- (The MR Enable field must be written with the appropriate value before this field is tested.)

MR-IOV Control Register Default Value (Offset 08h) — DWORD

(Switch)

- | | |
|--------------------------------|-------|
| a. Port Interrupt Enable | 0 |
| b. VS Interrupt Enable | 0 |
| c. Statistics Interrupt Enable | 0 |
| d. MR Switch Number | 0000h |

MR-IOV Status Register Default Value (Offset 0Ch) — DWORD

(Endpoint)

- | | |
|------------------|---|
| a. MSI Scheduled | 0 |
|------------------|---|

MR-IOV Status Register Default Value (Offset 0Ch) — DWORD

(Switch)

- | | |
|--------------------------------|---|
| a. Statistics Interrupt Status | 0 |
| b. MSI Scheduled | 0 |

MR-IOV VH Counts Register Default Value (Offset 10h) — DWORD

(Endpoint)

- a. NumVH

(if Is Main BF is 1 and MR Enable is written with 0) 00h

(The MR Enable field must be written with the appropriate value before this field is tested.)

MR-IOV Watchdog Timer Control Default Value Register (Offset 14h) — DWORD

(Switch)

- a. Watchdog Timer Interval

00h

MR-IOV VL Arbitration Capability and Status Default Value Register (Offset 20h) — DWORD

(Endpoint)

- a. VL Arbitration Status

0

MR-IOV VL Arbitration Control Default Value Register (Offset 24h) — DWORD

(Endpoint)

- a. VL Strict Priority Arbitration

00h

MR-IOV MR Error Control Default Value Register (Offset 2Eh) — WORD

(Endpoint)

- a. MR Uncorrectable Fatal TLP Error Mask
- b. MR Uncorrectable Global Key Error Mask

0

0

Function Table Entry (n) Function Control 1 Default Value ([table] + ([n] * [size])) + 08h — DWORD

(Endpoint)

- a. VC Extended VC Count 000b
- b. VC Low Priority Extended VC Count 000b
- c. MFVC Extended VC Count 000b
- d. MFVC Low Priority Extended VC Count 000b
- e. Global Key 000h
- f. Global Key Check Enable 0

(After returning the function to its default state test software must wait 1 second, then it must write the MR Enable field with 0, then the NumVH field must be written with the value returned by the MaxVH field, and then the MR Enable field must be written with 1 before this register is tested.)

Note: Testing this requires programming the designated Memory BAR and enabling Memory space. The value of [table] is the BAR contents (pointed to by the Function Table BIR field value) plus the offset (value in the Function Table Offset field multiplied by 8). The value of [size] is the value in the Function Table Entry Size field multiplied by 4. All values of [n] between 0 and the value in the MaxVH field inclusive must be tested.

**Function Table Entry (n) Function Control 2 Default Value ([table] + [n] * size) + 0Ch -
- DWORD**

(Endpoint)

a. VC Config Changed Interrupt Enable	0
b. MFVC Config Changed Interrupt Enable	0
c. PF Reset Initiated Interrupt Enable	0
d. VF Migration Status Interrupt Enable	0
e. VF Enable Interrupt Enable	0
f. VF Migration Capable	0
g. InitialVFs	
(for PFs with VF Mapping Supported as 1)	0000h
(for non-PFs)	0000h

(After returning the function to its default state test software must wait 1 second, then it must write the MR Enable field with 0, then the NumVH field must be written with the value returned by the MaxVH field, and then the MR Enable field must be written with 1 before this register is tested.)

Note: Testing this requires programming the designated Memory BAR and enabling Memory space. The value of [table] is the BAR contents (pointed to by the Function Table BIR field value) plus the offset (value in the Function Table Offset field multiplied by 8). The value of [size] is the value in the Function Table Entry Size field multiplied by 4. All values of [n] between 0 and the value in the MaxVH field inclusive must be tested.

**Function Table Entry (n) Function Control 3 Default Value ([table] + ([n] * [size])) + 10h
- DWORD**

(Endpoint)

a. TotalVFs	0000h
-------------	-------

(After returning the function to its default state test software must wait 1 second, then it must write the MR Enable field with 0, then the NumVH field must be written with the value returned by the MaxVH field, and then the MR Enable field must be written with 1 before this register is tested.)

Note: Testing this requires programming the designated Memory BAR and enabling Memory space. The value of [table] is the BAR contents (pointed to by the Function Table BIR field value) plus the offset (value in the Function Table Offset field multiplied by 8). The value of [size] is the value in the Function Table Entry Size field multiplied by 4. All values of [n] between 0 and the value in the MaxVH field inclusive must be tested.

Function Table Entry (n) Function Status Default Value ([table] + ([n] * [size])) + 14h — DWORD

(Endpoint)

- | | |
|------------------------------|---|
| a. VC Config Changed | 0 |
| b. MFVC Config Changed | 0 |
| c. PF Reset Initiated | 0 |
| d. VF Migration Status | 0 |
| e. VF Initialization Pending | 0 |

(After returning the function to its default state test software must wait 1 second, then it must write the MR Enable field with 0, then the NumVH field must be written with the value returned by the MaxVH field, and then the MR Enable field must be written with 1 before this register is tested.)

Note: Testing this requires programming the designated Memory BAR and enabling Memory space. The value of [table] is the BAR contents (pointed to by the Function Table BIR field value) plus the offset (value in the Function Table Offset field multiplied by 8). The value of [size] is the value in the Function Table Entry Size field multiplied by 4. All values of [n] between 0 and the value in the MaxVH field inclusive must be tested.

Function Table Entry (n) VC to VL Map 1 Default Value ([table] + ([n] * [size])) + 18h — DWORD

(Endpoint)

- | | |
|---|--------|
| a. VC0 VL Map
(if Is Main BF is 1 and MaxVL is non-zero
and VC Capability Supported is 1) | 000b |
| b. VC0 VL Map Enable
(if Is Main BF is 1 and MaxVL is non-zero
and VC Capability Supported is 1) | 0 or 1 |
| c. VC1 VL Map
(if Is Main BF is 1 and MaxVL is non-zero
and VC Capability Supported is 1
and Num VC Resources Hardware Present is 1 or greater) | 001b |
| d. VC1 VL Map Enable
(if Is Main BF is 1 and MaxVL is non-zero
and VC Capability Supported is 1
and Num VC Resources Hardware Present is 1 or greater) | 0 |
| e. VC2 VL Map
(if Is Main BF is 1 and MaxVL is non-zero
and VC Capability Supported is 1
and Num VC Resources Hardware Present is 2 or greater) | 010b |
| f. VC2 VL Map Enable
(if Is Main BF is 1 and MaxVL is non-zero
and VC Capability Supported is 1
and Num VC Resources Hardware Present is 2 or greater) | 0 |

- g. VC3 VL Map
(if Is Main BF is 1 and MaxVL is non-zero
and VC Capability Supported is 1
and Num VC Resources Hardware Present is 3 or greater) 011b
- h. VC3 VL Map Enable
(if Is Main BF is 1 and MaxVL is non-zero
and VC Capability Supported is 1
and Num VC Resources Hardware Present is 3 or greater) 0

(After returning the function to its default state test software must wait 1 second, then it must write the MR Enable field with 0, then the NumVH field must be written with the value returned by the MaxVH field, and then the MR Enable field must be written with 1 before this register is tested.)

Note: Testing this requires programming the designated Memory BAR and enabling Memory space. The value of [table] is the BAR contents (pointed to by the Function Table BIR field value) plus the offset (value in the Function Table Offset field multiplied by 8). The value of [size] is the value in the Function Table Entry Size field multiplied by 4. All values of [n] between 0 and the value in the MaxVH field inclusive must be tested.

Function Table Entry (n) VC to VL Map 2 Default Value ([table] + ([n] * [size])) + 1Ch — DWORD

(Endpoint)

- a. VC4 VL Map
(if Is Main BF is 1 and MaxVL is non-zero
and VC Capability Supported is 1
and Num VC Resources Hardware Present is 4 or greater) 100b
- b. VC4 VL Map Enable
(if Is Main BF is 1 and MaxVL is non-zero
and VC Capability Supported is 1
and Num VC Resources Hardware Present is 4 or greater) 0
- c. VC5 VL Map
(if Is Main BF is 1 and MaxVL is non-zero
and VC Capability Supported is 1
and Num VC Resources Hardware Present is 5 or greater) 101b
- d. VC5 VL Map Enable
(if Is Main BF is 1 and MaxVL is non-zero
and VC Capability Supported is 1
and Num VC Resources Hardware Present is 5 or greater) 0
- e. VC6 VL Map
(if Is Main BF is 1 and MaxVL is non-zero
and VC Capability Supported is 1
and Num VC Resources Hardware Present is 6 or greater) 110b
- f. VC6 VL Map Enable
(if Is Main BF is 1 and MaxVL is non-zero
and VC Capability Supported is 1
and Num VC Resources Hardware Present is 6 or greater) 0

- g. VC7 VL Map
(if Is Main BF is 1 and MaxVL is non-zero
and VC Capability Supported is 1
and Num VC Resources Hardware Present is 7 or greater) 111b
- h. VC7 VL Map Enable
(if Is Main BF is 1 and MaxVL is non-zero
and VC Capability Supported is 1
and Num VC Resources Hardware Present is 7 or greater) 0

(After returning the function to its default state test software must wait 1 second, then it must write the MR Enable field with 0, then the NumVH field must be written with the value returned by the MaxVH field, and then the MR Enable field must be written with 1 before this register is tested.)

Note: Testing this requires programming the designated Memory BAR and enabling Memory space. The value of [table] is the BAR contents (pointed to by the Function Table BIR field value) plus the offset (value in the Function Table Offset field multiplied by 8). The value of [size] is the value in the Function Table Entry Size field multiplied by 4. All values of [n] between 0 and the value in the MaxVH field inclusive must be tested.

Port Table Entry (n) Port Capability Default Value ([table] + ([n] * [size])) + 00h — DWORD

(Switch)

- a. Link Direction Supported
(if VS is Authorized and Non-PCIe Port (Management Port) is 1) 001b
- b. Port Direction Supported
(if VS is Authorized and Non-PCIe Port (Management Port) is 0) 01b, 10b, or 11b
(if VS is Authorized and Non-PCIe Port (Management Port) is 1) 10b

(After returning the function to its default state test software must wait 1 second, then it must write the MR Enable field with 0, then the NumVH field must be written with the value returned by the MaxVH field, and then the MR Enable field must be written with 1 before this register is tested.)

Note: Testing this requires programming the designated Memory BAR and enabling Memory space. The value of [table] is the BAR contents (pointed to by the Port Table BIR field value) plus the offset (value in the Port Table Offset field multiplied by 8). The value of [size] is the value in the Port Table Entry Size field multiplied by 4. All values of [n] between 0 and the value in the Num Port Table Entries field minus one inclusive must be tested.

**Port Table Entry (n) Port Control 1 Default Value ([table] + ([n] * [size])) + 04h —
DWORD**

(Switch)

- | | |
|---|---|
| a. Port DL_Up Interrupt Enable | 0 |
| b. Port DL_Down Interrupt Enable | 0 |
| c. Port PCIe Capability Interrupt Enable | 0 |
| d. Link Retrain Interrupt Enable | 0 |
| e. Beacon/WAKE# Interrupt Enable | 0 |
| f. MR Uncorrectable Fatal TLP Error Interrupt Enable | 0 |
| g. MR Uncorrectable Global Key Error Interrupt Enable | 0 |
| h. MR Correctable Global Key Error Interrupt Enable | 0 |
| i. Physical Hot-Plug Interrupt Enable | 0 |

(After returning the function to its default state test software must wait 1 second, then it must write the MR Enable field with 0, then the NumVH field must be written with the value returned by the MaxVH field, and then the MR Enable field must be written with 1 before this register is tested.)

Note: Testing this requires programming the designated Memory BAR and enabling Memory space. The value of [table] is the BAR contents (pointed to by the Port Table BIR field value) plus the offset (value in the Port Table Offset field multiplied by 8). The value of [size] is the value in the Port Table Entry Size field multiplied by 4. All values of [n] between 0 and the value in the Num Port Table Entries field minus one inclusive must be tested.

**Port Table Entry (n) Port Control 2 Default Value ([table] + ([n] * [size])) + 08h —
DWORD**

(Switch)

- | | |
|--|-----|
| a. VL Enable | |
| (if VS is Authorized and Non-PCIe Port (Management Port) is 0) | 01h |

(After returning the function to its default state test software must wait 1 second, then it must write the MR Enable field with 0, then the NumVH field must be written with the value returned by the MaxVH field, and then the MR Enable field must be written with 1 before this register is tested.)

Note: Testing this requires programming the designated Memory BAR and enabling Memory space. The value of [table] is the BAR contents (pointed to by the Port Table BIR field value) plus the offset (value in the Port Table Offset field multiplied by 8). The value of [size] is the value in the Port Table Entry Size field multiplied by 4. All values of [n] between 0 and the value in the Num Port Table Entries field minus one inclusive must be tested.

Port Table Entry (n) VL Arbitration Capability and Status Default Value ([table] + ([n] * [size])) + 14h — DWORD

(Switch)

- | | |
|--------------------------|---|
| a. VL Arbitration Status | 0 |
|--------------------------|---|

(After returning the function to its default state test software must wait 1 second, then it must write the MR Enable field with 0, then the NumVH field must be written with the value returned by the MaxVH field, and then the MR Enable field must be written with 1 before this register is tested.)

Note: Testing this requires programming the designated Memory BAR and enabling Memory space. The value of [table] is the BAR contents (pointed to by the Port Table BIR field value) plus the offset (value in the Port Table Offset field multiplied by 8). The value of [size] is the value in the Port Table Entry Size field multiplied by 4. All values of [n] between 0 and the value in the Num Port Table Entries field minus one inclusive must be tested.

Port Table Entry (n) VL Arbitration Control Default Value ([table] + ([n] * [size])) + 18h — DWORD

(Switch)

- | | |
|-----------------------------------|-----|
| a. Load VL Arbitration Table | 0 |
| b. VL Strict Priority Arbitration | 00h |

(After returning the function to its default state test software must wait 1 second, then it must write the MR Enable field with 0, then the NumVH field must be written with the value returned by the MaxVH field, and then the MR Enable field must be written with 1 before this register is tested.)

Note: Testing this requires programming the designated Memory BAR and enabling Memory space. The value of [table] is the BAR contents (pointed to by the Port Table BIR field value) plus the offset (value in the Port Table Offset field multiplied by 8). The value of [size] is the value in the Port Table Entry Size field multiplied by 4. All values of [n] between 0 and the value in the Num Port Table Entries field minus one inclusive must be tested.

Port Table Entry (n) MR Error Control Default Value ([table] + ([n] * [size])) + 22h — WORD

(Switch)

- | | |
|---|---|
| a. MR Uncorrectable Fatal TLP Error Mask | 0 |
| b. MR Uncorrectable Global Key Error Mask | 0 |
| c. MR Correctable Global Key Error Mask | 0 |

(After returning the function to its default state test software must wait 1 second, then it must write the MR Enable field with 0, then the NumVH field must be written with the value returned by the MaxVH field, and then the MR Enable field must be written with 1 before this register is tested.)

Note: Testing this requires programming the designated Memory BAR and enabling Memory space. The value of [table] is the BAR contents (pointed to by the Port Table BIR field value) plus the offset (value in the Port Table Offset field multiplied by 8). The value of [size] is the value in the Port Table Entry Size field multiplied by 4. All values of [n] between 0 and the value in the Num Port Table Entries field minus one inclusive must be tested.

Port Table Entry (n) PCI Express Capabilities Default Value ([table] + ([n] * [size])) + [offset] + 02h — WORD

(Switch if PCIe Offset is non-zero)

- | | |
|-----------------------|----|
| a. Capability Version | 2h |
|-----------------------|----|

(After returning the function to its default state test software must wait 1 second, then it must write the MR Enable field with 0, then the NumVH field must be written with the value returned by the MaxVH field, and then the MR Enable field must be written with 1 before this register is tested.)

Note: Testing this requires programming the designated Memory BAR and enabling Memory space. The value of [table] is the BAR contents (pointed to by the Port Table BIR field value) plus the offset (value in the Port Table Offset field multiplied by 8). The value of [size] is the value in the Port Table Entry Size field multiplied by 4. All values of [n] between 0 and the value in the Num Port Table Entries field minus one inclusive must be tested. The value of [offset] is the value in the PCIe Offset field multiplied by 4. If the PCIe Offset field is zero, then this register does not exist and is not tested.

Port Table Entry (n) Link Control Default Value ([table] + ([n] * [size])) + [offset] + 10h - WORD

(Switch if PCIe Offset is non-zero)

- | | |
|---|---|
| a. Link Disable | 0 |
| b. Common Clock Configuration | 0 |
| c. Extended Synch | 0 |
| d. Enable Clock Power Management | 0 |
| e. Hardware Autonomous Width Disable | 0 |
| f. Link Bandwidth Management Interrupt Enable | 0 |
| g. Link Autonomous Bandwidth Interrupt Enable | 0 |

(After returning the function to its default state test software must wait 1 second, then it must write the MR Enable field with 0, then the NumVH field must be written with the value returned by the MaxVH field, and then the MR Enable field must be written with 1 before this register is tested.)

Note: Testing this requires programming the designated Memory BAR and enabling Memory space. The value of [table] is the BAR contents (pointed to by the Port Table BIR field value) plus the offset (value in the Port Table Offset field multiplied by 8). The value of [size] is the value in the Port Table Entry Size field multiplied by 4. All values of [n] between 0 and the value in the Num Port Table Entries field minus one inclusive must be tested. The value of [offset] is the value in the PCIe Offset field multiplied by 4. If the PCIe Offset field is zero, then this register does not exist and is not tested.

Port Table Entry (n) Slot Control Default Value ([table] + ([n] * [size])) + [offset] + 18h - WORD

(Switch if PCIe Offset is non-zero)

a. Attention Button Pressed Enable	0
b. Power Fault Detected Enable	0
c. MRL Sensor Changed Enable	0
d. Presence Detect Changed Enable	0
e. Command Completed Interrupt Enable	0
f. Hot-Plug Interrupt Enable	0
g. Data Link Layer State Changed Enable	0

(After returning the function to its default state test software must wait 1 second, then it must write the MR Enable field with 0, then the NumVH field must be written with the value returned by the MaxVH field, and then the MR Enable field must be written with 1 before this register is tested.)

Note: Testing this requires programming the designated Memory BAR and enabling Memory space. The value of [table] is the BAR contents (pointed to by the Port Table BIR field value) plus the offset (value in the Port Table Offset field multiplied by 8). The value of [size] is the value in the Port Table Entry Size field multiplied by 4. All values of [n] between 0 and the value in the Num Port Table Entries field minus one inclusive must be tested. The value of [offset] is the value in the PCIe Offset field multiplied by 4. If the PCIe Offset field is zero, then this register does not exist and is not tested.

Port Table Entry (n) Slot Status Default Value ([table] + ([n] * [size])) + [offset] + 1Ah - WORD

(Switch if PCIe Offset is non-zero)

a. Attention Button Pressed	0
b. Power Fault Detected	0
c. MRL Sensor Changed	0
d. Presence Detect Changed	0
e. Command Completed	0

(After returning the function to its default state test software must wait 1 second, then it must write the MR Enable field with 0, then the NumVH field must be written with the value returned by the MaxVH field, and then the MR Enable field must be written with 1 before this register is tested.)

Note: Testing this requires programming the designated Memory BAR and enabling Memory space. The value of [table] is the BAR contents (pointed to by the Port Table BIR field value) plus the offset (value in the Port Table Offset field multiplied by 8). The value of [size] is the value in the Port Table Entry Size field multiplied by 4. All values of [n] between 0 and the value in the Num Port Table Entries field minus one inclusive must be tested. The value of [offset] is the value in the PCIe Offset field multiplied by 4. If the PCIe Offset field is zero, then this register does not exist and is not tested.

Port Table Entry (n) Device Control 2 Default Value ([table] + ([n] * [size])) + [offset] + 20h — WORD

(Switch if PCIe Offset is non-zero)

- | | |
|---------------------------------|-----|
| a. AtomicOp Egress Blocking | 0 |
| b. LTR Mechanism Enable | 0 |
| c. OBFF Enable | |
| (For Base 2.x or later testing) | 00b |
| d. End-End TLP Prefix Blocking | 0 |

(After returning the function to its default state test software must wait 1 second, then it must write the MR Enable field with 0, then the NumVH field must be written with the value returned by the MaxVH field, and then the MR Enable field must be written with 1 before this register is tested.)

Note: Testing this requires programming the designated Memory BAR and enabling Memory space. The value of [table] is the BAR contents (pointed to by the Port Table BIR field value) plus the offset (value in the Port Table Offset field multiplied by 8). The value of [size] is the value in the Port Table Entry Size field multiplied by 4. All values of [n] between 0 and the value in the Num Port Table Entries field minus one inclusive must be tested. The value of [offset] is the value in the PCIe Offset field multiplied by 4. If the PCIe Offset field is zero, then this register does not exist and is not tested.

Port Table Entry (n) Link Control 2 Default Value ([table] + ([n] * [size])) + [offset] + 28h — WORD

(Switch if PCIe Offset is non-zero)

- | | |
|---------------------------------|----------------|
| a. Target Link Speed | |
| (For Base 2.x or later testing) | [HLS] |
| | see Note below |

Note: The Highest supported speed value [HLS] is the value returned in the Port Table Entry Max Link Speed/Supported Link Speeds. [HLS] may also be 0000b if Max Link Speed/Supported Link Speeds returns 0001b (2.5 GT/s only).

Note: This test must only write supported values as reported in the Port Table Entry Max Link Speed/Supported Link Speeds field.

- | | |
|--|-------|
| b. Hardware Autonomous Speed Disable | 0 |
| c. Transmit Margin | 000b |
| d. Enter Modified Compliance | 0 |
| e. Compliance SOS | 0 |
| f. Compliance Preset/De-emphasis | |
| (For Base 3.x or later testing) | 0000b |
| Note: For Base 3.x or later, this field consists of bits 15-12 of this register. | |
| g. Compliance De-emphasis | |
| (For Base 2.x testing) | 0 |

Note: For Base 2.x, this field consists of bit 12 of this register.

(After returning the function to its default state test software must wait 1 second, then it must write the MR Enable field with 0, then the NumVH field must be written with the value returned by the MaxVH field, and then the MR Enable field must be written with 1 before this register is tested.)

Note: Testing this requires programming the designated Memory BAR and enabling Memory space. The value of [table] is the BAR contents (pointed to by the Port Table BIR field value)

plus the offset (value in the Port Table Offset field multiplied by 8). The value of [size] is the value in the Port Table Entry Size field multiplied by 4. All values of [n] between 0 and the value in the Num Port Table Entries field minus one inclusive must be tested. The value of [offset] is the value in the PCIe Offset field multiplied by 4. If the PCIe Offset field is zero, then this register does not exist and is not tested.

Port Table Entry (n) Link Status 2 Default Value ([table] + ([n] * [size])) + [offset] + 2Ah — WORD

(Switch if PCIe Offset is non-zero)

a. Link Equalization Request

(For Base 3.x or later testing:

if Max Link Speed/Supported Link Speeds is 0010b or less) 0

Note: A function under test that supports 8.0 GT/s will participate in Link Equalization, so the default values of the equalization status fields cannot be tested on an 8.0 GT/s capable device.

(After returning the function to its default state test software must wait 1 second, then it must write the MR Enable field with 0, then the NumVH field must be written with the value returned by the MaxVH field, and then the MR Enable field must be written with 1 before this register is tested.)

Note: Testing this requires programming the designated Memory BAR and enabling Memory space. The value of [table] is the BAR contents (pointed to by the Port Table BIR field value) plus the offset (value in the Port Table Offset field multiplied by 8). The value of [size] is the value in the Port Table Entry Size field multiplied by 4. All values of [n] between 0 and the value in the Num Port Table Entries field minus one inclusive must be tested. The value of [offset] is the value in the PCIe Offset field multiplied by 4. If the PCIe Offset field is zero, then this register does not exist and is not tested.

VS Table Entry (n) VS Control Default Value ([table] + ([n] * [size])) + 04h — DWORD

(Switch)

- | | |
|---|------|
| a. VS Bridge Interrupt Enable | 0 |
| b. VS Global Key Value | 000h |
| c. VS Global Key Entering Check Enable | 0 |
| d. VS Global Key Exiting Check Enable | 0 |
| e. VS Global Key Terminating Check Enable | 0 |

(After returning the function to its default state test software must wait 1 second, then it must write the MR Enable field with 0, then the NumVH field must be written with the value returned by the MaxVH field, and then the MR Enable field must be written with 1 before this register is tested.)

Note: Testing this requires programming the designated Memory BAR and enabling Memory space. The value of [table] is the BAR contents (pointed to by the VS Table BIR field value) plus the offset (value in the VS Table Offset field multiplied by 8). The value of [size] is the value in the VS Table Entry Size field multiplied by 4. All values of [n] between 0 and the value in the Num VS Table Entries field minus one inclusive must be tested.

VS Bridge Table Entry (n, x) VS Bridge Capability and Status Default Value ([table] + (([n] * [bsize]) + ([x] * [size]))) + 00h — DWORD

(Switch)

- | | |
|--------------------------------------|---|
| a. PME Turn Off State Changed | 0 |
| b. Power Controller State Changed | 0 |
| c. Power Indicator State Changed | 0 |
| d. Attention Indicator State Changed | 0 |
| e. VC Config Changed | 0 |

(After returning the function to its default state test software must wait 1 second, then it must write the MR Enable field with 0, then the NumVH field must be written with the value returned by the MaxVH field, and then the MR Enable field must be written with 1 before this register is tested.)

Note: Testing this requires programming the designated Memory BAR and enabling Memory space. The value of [table] is the BAR contents (pointed to by the VS Bridge Table BIR field value) plus the offset (value in the VS Bridge Table Offset field multiplied by 8). The value of [bsize] is the value in the Num VS Bridge Table Entries field multiplied by the VS Bridge Table Entry Size field multiplied by 4. The value of [size] is the value in the VS Bridge Table Entry Size field multiplied by 4. All values of [x] between 0 and the value in the Num VS Bridge Table Entries field minus one inclusive must be tested. All values of [n] between 0 and the value in the Num VS Table Entries field minus one inclusive must be tested.

VS Bridge Table Entry (n, x) VS Bridge Control 1 Default Value ([table] + (([n] * [bsize]) + ([x] * [size]))) + 04h — DWORD

(Switch)

- | | |
|---|---|
| a. VC Config Changed Interrupt Enable | 0 |
| b. PME Turn Off State Change Interrupt Enable | 0 |

(After returning the function to its default state test software must wait 1 second, then it must write the MR Enable field with 0, then the NumVH field must be written with the value returned by the MaxVH field, and then the MR Enable field must be written with 1 before this register is tested.)

Note: Testing this requires programming the designated Memory BAR and enabling Memory space. The value of [table] is the BAR contents (pointed to by the VS Bridge Table BIR field value) plus the offset (value in the VS Bridge Table Offset field multiplied by 8). The value of [bsize] is the value in the Num VS Bridge Table Entries field multiplied by the VS Bridge Table Entry Size field multiplied by 4. The value of [size] is the value in the VS Bridge Table Entry Size field multiplied by 4. All values of [x] between 0 and the value in the Num VS Bridge Table Entries field minus one inclusive must be tested. All values of [n] between 0 and the value in the Num VS Table Entries field minus one inclusive must be tested.

VS Bridge Table Entry (n, x) Virtual Hot-Plug Signals Interface 2 Default Value ([table] + (([n] * [bsize]) + ([x] * [size]))) + 10h — DWORD

(Switch)

- | | |
|--|---|
| a. Virtual Slot Implemented | 0 |
| b. Hot Plug Signals Interrupt Enable | 0 |
| c. Virtual Discard Ingress Request | 0 |
| d. Virtual Presence Detect State
(if VS is Authorized and Hot-Plug Hardware Present is 1) | 0 |
| e. Virtual Force Reset | 0 |

(After returning the function to its default state test software must wait 1 second, then it must write the MR Enable field with 0, then the NumVH field must be written with the value returned by the MaxVH field, and then the MR Enable field must be written with 1 before this register is tested.)

Note: Testing this requires programming the designated Memory BAR and enabling Memory space. The value of [table] is the BAR contents (pointed to by the VS Bridge Table BIR field value) plus the offset (value in the VS Bridge Table Offset field multiplied by 8). The value of [bsize] is the value in the Num VS Bridge Table Entries field multiplied by the VS Bridge Table Entry Size field multiplied by 4. The value of [size] is the value in the VS Bridge Table Entry Size field multiplied by 4. All values of [x] between 0 and the value in the Num VS Bridge Table Entries field minus one inclusive must be tested. All values of [n] between 0 and the value in the Num VS Table Entries field minus one inclusive must be tested.

12. For functions under test that have a link, the test is run at each of the following link speeds:
 - a. 2.5 GT/s (for devices that are capable). See Section 2.1.2.12 for details.
 - b. 5.0 GT/s (for devices that are capable). See Section 2.1.2.13 for details.
 - c. 8.0 GT/s (for devices that are capable). See Section 2.1.2.14 for details.
13. If the device type is Root Port or Root Complex Integrated Endpoint, then the test is repeated using each RCRB associated with the function under test, or associated with an RCRB that is itself associated with the function under test. See Section 2.1.2.17 for details. All offsets in the tests are now relative to the RCRB's base address and all cycles are now memory space accesses.

The test *fails* if:

- ☐ More than one MR-IOV capability structure is present.
- ☐ An RCRB contains a MR-IOV capability structure.
- ☐ The Capability Version field does not report 1h.
- ☐ The Next Capability Offset field does not report 000h or a value greater than 0FFh or the lower two bits are not 00b.
- ☐ The MR-IOV capability structure is present in a device that is not an Endpoint or a Switch.
- ☐ The MR-IOV capability structure is present in a VF.
- ☐ For an Endpoint, the implemented VL Enable fields are not RW when MR Enable is 1.
- ☐ For an Endpoint, the implemented VL Enable fields are not RO when MR Enable is 0.
- ☐ For an Endpoint, the NumVH field is not RW when MR Enable is 0.
- ☐ For an Endpoint, the NumVH field is not RO when MR Enable is 1.
- ☐ For an Endpoint, the Function Table Entry Size field is less than 16.

- ☐ For an Endpoint, the Function Table BIR field contains a reserved value.
- ☐ For an Endpoint, the Function Table BIR field value points to a non-memory BAR.
- ☐ For an Endpoint, the Function Table Offset field is less than 4.
- ☐ For an Endpoint, the LVF Table BIR field returns a reserved value.
- ☐ For an Endpoint, the LVF Table BIR field value points to a non-memory BAR.
- ☐ For an Endpoint that has the VL Arbitration Present field as 1, the VL Arbitration Capability field returns a reserved value.
- ☐ For an Endpoint that has the VL Arbitration Present field as 1 and the VL Arbitration Capability field supports Time-based WRR arbitration, the Reference Clock field returns a reserved value.
- ☐ For a Switch, the VS Authorization Bitmap Offset field returns a value less than 100h.
- ☐ For a Switch, the Port Table Entry Size field is less than 31.
- ☐ For a Switch, the Port Table BIR field contains a reserved value.
- ☐ For a Switch, the Port Table BIR field value points to a non-memory BAR.
- ☐ For a Switch, the Port Table Offset field is less than 4.
- ☐ For a Switch, the VS Table Entry Size field is less than 3.
- ☐ For a Switch, the VS Table BIR field contains a reserved value.
- ☐ For a Switch, the VS Table BIR field value points to a non-memory BAR.
- ☐ For a Switch, the VS Table Offset field is less than 4.
- ☐ For a Switch, the VS Bridge Table Entry Size field is less than 12.
- ☐ For a Switch, the VS Bridge Table BIR field contains a reserved value.
- ☐ For a Switch, the VS Bridge Table BIR field value points to a non-memory BAR.
- ☐ If VL Arbitration is supported, the VL Arbitration Table BIR field contains a reserved value.
- ☐ If VL Arbitration is supported, the VL Arbitration Table BIR field value points to a non-memory BAR.
- ☐ The Number of Statistics Blocks field returns a value greater than 31.
- ☐ Unimplemented bits in the Statistics Block Start/Busy register always return 0.
- ☐ Unimplemented bits in the Statistics Block Stop/Busy register always return 0.
- ☐ If the Number of Statistics Descriptors field is non-zero, the Statistics Descriptor Table BIR field contains a reserved value.
- ☐ If the Number of Statistics Descriptors field is non-zero, the Statistics Descriptor Table BIR field value points to a non-memory BAR.
- ☐ If the Number of Statistics Blocks field is non-zero, the Statistics Block Table BIR field contains a reserved value.
- ☐ If the Number of Statistics Blocks field is non-zero, the Statistics Block Table BIR field value points to a non-memory BAR.
- ☐ Any of the register field characteristic tests fail.
- ☐ Any of the default value tests fail.

B

APPENDIX B. Obsolete Test Details

Tests that are no longer supported in this version of the test document are listed below for historical reference purposes. Existing test numbers should only be re-used in the event that the test is re-instated, or a new test is developed that overlaps the coverage of the obsolete test.

B.1 TD_1_1

B.2 TD_1_33 Root Complex Integrated Endpoint

B.3 TD_1_35x Configuration Access Correlation Extended Capability Structure

Test TD_1_35x is now replaced by TD_1_35, as this capability structure is no longer supported by the current Base specification. The test description below applies to a previous version of the Base specification, when the capability structure was supported, and is included only for historical reasons.

The test verifies that if the function under test reports a PCI Express Configuration Access Correlation Extended Capability structure, it is implemented as defined in the relevant specifications. (A Configuration Access Correlation Extended Capability structure can only be implemented in a Base 1.1 device. Starting with Base 2.0, a Configuration Access Correlation Extended Capability structure is no longer supported and must not be implemented.)

Relevant Specifications

- ☐ *PCI Express Base Specification*
- ☐ *ECN Trusted Configuration Space (to Base 1.1)*

(See Section 1.3 for additional specification revision details.)

Applicable Device/Port Types

This test is run on all device/port types.

This test is run on any RCRB associated with the function under test.

Starting Configuration

This test is run with functions starting in the D0-Uninitialized state following the standard initialization procedures in Section 2.1.1. Testing in additional Dx states may be optionally supported. ASPM states of the upstream and downstream ports are disabled for this test.

Overview of Test Steps

Test software performs the following steps:

1. Configure the function under test following the procedure described in Section 2.1.1.
2. Examine the Extended Capability ID field for each of the detected Extended Capabilities. Determine how many instances of the Extended Capability ID of 000Ch (Configuration Access Correlation Extended Capability) are found. If more than one is found, the test terminates with a failure.
3. If the Extended Capability ID of 000Ch is found in an RCRB, the test terminates with a failure.
4. For Base 2.x or later testing: if the Extended Capability ID of 000Ch is found, the test terminates with a failure.
5. For Base 1.x testing: if an Extended Capability ID of 000Ch is found for an extended capability the following checks are performed on that extended capability structure:
6. The Capability Version field must be 1h.
7. The Next Capability Offset field must be 000h or greater than 0FFh and the lower 2 bits of this field must be 00b.
8. The following register field characteristic checks are performed:

Configuration Access Correlation Extended Capability Header (Offset 00 — DWORD)

- | | |
|---------------------------|----|
| a. Extended Capability ID | RO |
| b. Capability Version | RO |
| c. Next Capability Offset | RO |

Device Correlation Register (Offset 04 — DWORD)

(all bits) RO

9. For functions under test that have a link, the test is run at each of the following link speeds:
 - a. 2.5 GT/s (for devices that are capable). See Section 2.1.2.12 for details.
 - b. 5.0 GT/s (for devices that are capable). See Section 2.1.2.13 for details.
 - c. 8.0 GT/s (for devices that are capable). See Section 2.1.2.14 for details.
10. If the device type is Root Port or Root Complex Integrated Endpoint, then the test is repeated using each RCRB associated with the function under test, or associated with an RCRB that is itself associated with the function under test. See Section 2.1.2.17 for details. All offsets in the tests are now relative to the RCRB's base address and all cycles are now memory space accesses.

The test *fails* if:

- ☐ More than one Configuration Access Correlation capability structure is present.
- ☐ An RCRB contains a Configuration Access Correlation capability structure.
- ☐ The Configuration Access Correlation capability structure is present in any function (for Base 2.x or later testing only).

- ❑ The Capability Version field does not report 1h (for Base 1.x testing only).
- ❑ The Next Capability Offset field does not report 000h or a value greater than 0FFh or the lower two bits are not 00b (for Base 1.x testing only).
- ❑ Any of the register field characteristic tests fail.

B.4 TD_1_36 Trusted Config Space Header

Test TD_1_36 is no longer supported, as this trusted capability structure is no longer supported by the current Base specification. The test description below applies to a previous version of the Base specification, when the capability structure was supported, and is included only for historical reasons.

The test verifies that if the function under test reports support for Trusted Configuration Space, it implements a Trusted Configuration Space Header as defined in the relevant specifications.

The test requires a platform that supports Trusted Configuration Cycles.

Relevant Specifications

- ❑ *ECN Trusted Configuration Space (to Base 1.1)*

(See Section 1.3 for additional specification revision details.)

Applicable Device/Port Types

This test is run on all device/port types implementing Trusted Configuration Space.

Starting Configuration

This test is run with functions starting in the D0-Uninitialized state following the standard initialization procedures in Section 2.1.1. Testing in additional Dx states may be optionally supported. ASPM states of the upstream and downstream ports are disabled for this test.

Overview of Test Steps

Test software performs the following steps:

1. Configure the function under test following the procedure described in Section 2.1.1.
2. For Base 1.x testing: read the 12 bits from location 004h (First Trusted Capability Offset field) in the Trusted Configuration Space and perform the following checks:
 - a. The lower two bits must be zero (00b).
 - b. The value read, if non-zero, must be greater than 007h and less than FFDh.
3. The following register field characteristic checks are performed:

Trusted Configuration Register 1 (Location 00h) — DWORD

Trusted Class Code	RO
RsvdP_31-24	RO-Zero

Trusted Configuration Register 2 (Location 04h) — DWORD

First Trusted Capability Offset	RO
RsvdP_31-12	RO-Zero

4. For functions under test that have a link, the test is run at each of the following link speeds:
 - a. 2.5 GT/s (for devices that are capable). See Section 2.1.2.12 for details.
 - b. 5.0 GT/s (for devices that are capable). See Section 2.1.2.13 for details.
 - c. 8.0 GT/s (for devices that are capable). See Section 2.1.2.14 for details.

The test *fails* if:

- ☐ The First Trusted Capability Offset field does not report 000h or a value greater than 007h or the lower two bits are not 00b (for Base 1.x testing only).
- ☐ Any of the register field characteristic tests fail.

B.5 TD_1_37 Configuration Access Correlation Trusted Capability Structure

Test TD_1_37 is no longer supported, as this trusted capability structure is no longer supported by the current Base specification. The test description below applies to a previous version of the Base specification, when the capability structure was supported, and is included only for historical reasons.

The test verifies that if the function under test reports a PCI Express Configuration Access Correlation Trusted Capability structure, it is implemented as defined in the relevant specifications. (A Configuration Access Correlation Trusted Capability structure can only be implemented in a Base 1.1 device. Starting with Base 2.0, a Configuration Access Correlation Trusted Capability structure is no longer supported and must not be implemented.)

The test requires a platform that supports Trusted Configuration Cycles.

Relevant Specifications

- ☐ *ECN Trusted Configuration Space (to Base 1.1)*

(See Section 1.3 for additional specification revision details.)

Applicable Device/Port Types

This test is run on all device/port types implementing Trusted Configuration Space.

This test is run on any RCRB associated with the function under test.

Starting Configuration

This test is run with functions starting in the D0-Uninitialized state following the standard initialization procedures in Section 2.1.1. Testing in additional Dx states may be optionally supported. ASPM states of the upstream and downstream ports are disabled for this test.

Overview of Test Steps

Test software performs the following steps:

1. Configure the function under test following the procedure described in Section 2.1.1.
2. Examine the Trusted Capability ID field for each of the detected Trusted Capabilities. Determine how many instances of the Trusted Capability ID of 0001h (Configuration Access Correlation Trusted Capability) are found. If more than one is found, the test terminates with a failure.
3. If the Trusted Capability ID of 0001h is found in an RCRB, the test terminates with a failure.

4. For Base 2.x or later testing: if the Trusted Capability ID of 0001h is found, the test terminates with a failure.
5. For Base 1.x testing: if a Trusted Capability ID of 0001h is found for a trusted capability the following checks are performed on that trusted capability structure:
6. The Trusted Capability Version field must be 1h.
7. The Next Trusted Capability Offset field must be 000h or greater than 007h and the lower 2 bits of this field must be 00b.
8. The following register field characteristic checks are performed:

Configuration Access Correlation Trusted Capability Header (Offset 00 — DWORD)

- | | |
|-----------------------------------|----|
| a. Trusted Capability ID | RO |
| b. Trusted Capability Version | RO |
| c. Next Trusted Capability Offset | RO |

Device Correlation Register (Offset 04 — DWORD)

(all bits)	RO
------------	----

9. For functions under test that have a link, the test is run at each of the following link speeds:
 - a. 2.5 GT/s (for devices that are capable). See Section 2.1.2.12 for details.
 - b. 5.0 GT/s (for devices that are capable). See Section 2.1.2.13 for details.
 - c. 8.0 GT/s (for devices that are capable). See Section 2.1.2.14 for details.
10. If the device type is Root Port or Root Complex Integrated Endpoint, then the test is repeated using each RCRB associated with the function under test, or associated with an RCRB that is itself associated with the function under test. See Section 2.1.2.17 for details. All offsets in the tests are now relative to the RCRB's base address and all cycles are now memory space accesses.

The test *fails* if:

- ☐ More than one Configuration Access Correlation trusted capability structure is present.
- ☐ An RCRB contains a Configuration Access Correlation trusted capability structure.
- ☐ The Configuration Access Correlation trusted capability structure is present in any function (for Base 2.x or later testing only).
- ☐ The Trusted Capability Version field does not report 1h (for Base 1.x testing only).
- ☐ The Next Trusted Capability Offset field does not report 000h or a value greater than 007h or the lower two bits are not 00b (for Base 1.x testing only).
- ☐ Any of the register field characteristic tests fail.

B.6 TD_1_38 Vendor-Specific Trusted Capability Structure

Test TD_1_38 is no longer supported, as this trusted capability structure is no longer supported by the current Base specification. The test description below applies to a previous version of the Base specification, when the capability structure was supported, and is included only for historical reasons.

The test verifies that if a function under test reports a PCI Express Vendor-Specific Trusted Capability structure, it is implemented as defined in the relevant specifications.

The test requires a platform that supports Trusted Configuration Cycles.

Relevant Specifications

❑ *ECN Trusted Configuration Space (to Base 1.1)*

(See Section 1.3 for additional specification revision details.)

Applicable Device/Port Types

This test is run on all device/port types implementing Trusted Configuration Space.

This test is run on any RCRB associated with the function under test.

Starting Configuration

This test is run with functions starting in the D0-Uninitialized state following the standard initialization procedures in Section 2.1.1. Testing in additional Dx states may be optionally supported. ASPM states of the upstream and downstream ports are disabled for this test.

Overview of Test Steps:

Test software performs the following steps:

1. Configure the function under test following the procedure described in Section 2.1.1.
2. Examine the Trusted Capability ID field for each of the detected Trusted Capabilities. Determine how many instances of the Trusted Capability ID of 0002h (Vendor-Specific Trusted Capability) are found.
3. If the Trusted Capability ID of 0002h is found in an RCRB, the test terminates with a failure.
4. If a Trusted Capability ID of 0002h is found for a trusted capability the following checks are performed for each instance of that trusted capability structure:
5. The Trusted Capability Version field must be 1h.
6. The Next Trusted Capability Offset field must be 000h or greater than 007h and the lower 2 bits of this field must be 00b.
7. The Next Trusted Capability Offset field must return a value less than the offset of this capability by at least 12 bytes or must be larger than the offset of this capability plus the VSTC Length field value rounded to the next highest DWORD boundary.
8. The value for the VSTC Length field must be at least 00Ch (12 bytes).
9. If the VSTC Length field is greater than 00Ch, test software reads each byte starting at offset 0Ch, and continuing until the byte length given by the value in the VSTC Length field is reached. (This checks that Vendor-Specific registers can be read safely. No check on the returned data value is done.)

10. The following register field characteristic checks are performed:

Vendor-Specific Capability Trusted Capability Header (Offset 00h) — DWORD

- | | |
|--------------------------------------|----|
| a. PCI Express Trusted Capability ID | RO |
| b. Trusted Capability Version | RO |
| c. Next Trusted Capability Offset | RO |

Vendor-Specific Header (Offset 04h) — DWORD

- | | |
|----------------|----|
| a. VSTC ID | RO |
| b. VSTC Rev | RO |
| c. VSTC Length | RO |

VSTC Vendor ID (Offset 08h) — DWORD

- | | |
|-------------------|---------|
| a. VSTC Vendor ID | RO |
| b. RsvdP_31-16 | RO-Zero |

11. For functions under test that have a link, the test is run at each of the following link speeds:

- a. 2.5 GT/s (for devices that are capable). See Section 2.1.2.12 for details.
- b. 5.0 GT/s (for devices that are capable). See Section 2.1.2.13 for details.
- c. 8.0 GT/s (for devices that are capable). See Section 2.1.2.14 for details.

12. If the device type is Root Port or Root Complex Integrated Endpoint, then the test is repeated using each RCRB associated with the function under test, or associated with an RCRB that is itself associated with the function under test. See Section 2.1.2.17 for details. All offsets in the tests are now relative to the RCRB's base address and all cycles are now memory space accesses.

The test *fails* if:

- ☐ An RCRB contains a Vendor Specific trusted capability structure.
- ☐ The Trusted Capability Version field does not report 1h.
- ☐ The Next Trusted Capability Offset field does not report 000h or a value greater than 007h or the lower two bits are not 00b.
- ☐ The Next Trusted Capability Offset field is not at least 12 less than the current capability offset or it is less than the current capability offset plus the VSTC Length field value rounded to the next highest DWORD boundary.
- ☐ The VSTC Length field is less than 12.
- ☐ Any of the register field characteristic tests fail.

B.7 TD_3_3 Accurate Slot Reporting

Test TD_3_3 is no longer supported as it requires the system to reboot while the test is running, which the test cannot support. Also it requires the removal of all PCI Express cards, which cannot be done if some cards are required to support the test itself (such as a display card).

The test verifies that the system under test can accurately report physically available slots (those that are not used by system-integrated devices or are not exposed) versus silicon supported ports that are not connected to physical slots. (The test prompts the user to unpopulate/populate the physical slots under test.)

Relevant Specifications

❑ *PCI Express Base Specification*

(See Section 1.3 for additional specification revision details.)

Applicable Device/Port Types

This test is run on downstream ports of Root Ports, Switch Downstream Ports, and PCI/PCI-X to PCI Express Bridges.

Starting Configuration

For a non-Root Port function under test, it is completely uninitialized following a reset, except that the bus number fields are initialized with valid numbers.

The immediate downstream device (the upstream port of the link connected to the function under test) is completely uninitialized following a reset.

Overview of Test Steps

Test software performs the following steps:

1. Prompt the user to power down (if necessary) and remove all PCI Express cards from physically exposed slots.
2. After a system reboot (if necessary) test software examines the PCI Express Capabilities register for each Downstream Port in the system.
3. For each Downstream Port that has a PCI Express Capabilities register that has the Slot Implemented field returns 1:
 - a. Test software checks the Presence Detect State field of the Slot Status register. If the Presence Detect State field returns 1 the test fails.
 - b. Test software checks the Presence Detect State field of the Slot Status register. If the Presence Detect State field returns 0 record this as an implemented slot on a Downstream Port that does not have a card present.
4. Prompt the user to power down (if necessary) and populate with PCI Express cards each of the physically exposed slots.

5. After a system reboot (if necessary) test software does the following for each of the previously unpopulated Downstream Ports that has a PCI Express Capabilities register that has the Slot Implemented field set to 1 (using the information recorded in step 3):
 - a. Test software checks the Presence Detect State field of the Slot Status register. If the Presence Detect State field returns 0 the test fails.

The test *fails* if:

- ☐ A populated physical slot cannot be unpopulated by the user (as indicated by the Presence Detect State field of the Slot Status register returning 1).
- ☐ An unpopulated physical slot cannot be populated by the user (as indicated by the Presence Detect State field of the Slot Status register returning 0).

B.8 TD_3_4 Basic Hot-Plug Insertion

Test TD_3_4 is no longer supported as it requires the removal of all PCI Express cards, which cannot be done if some cards are required to support the test itself (such as a display card).

The test verifies that the system under test can properly handle hot-plug situations of hot insert and that each downstream port connected to a physical slot accurately reports its hot-plug state in its Slot Capabilities, Slot Control, and Slot Status registers. (The test prompts the user to unpopulate/populate the hot-plug slots under test. If an Attention Button is implemented, the test also prompts the user to press it.)

Relevant Specifications

- ☐ *PCI Express Base Specification*

(See Section 1.3 for additional specification revision details.)

Applicable Device/Port Types

This test is run on downstream ports of Root Ports, Switch Downstream Ports, and PCI/PCI-X to PCI Express Bridges.

Starting Configuration

For a non-Root Port function under test, it is completely uninitialized following a reset, except that the bus number fields are initialized with valid numbers.

The immediate downstream device (the upstream port of the link connected to the function under test) is completely uninitialized following a reset.

Overview of Test Steps

Test software performs the following steps:

1. Prompt the user to power down (if necessary) and remove all PCI Express cards from physically exposed slots (hot-plug slots and optionally non-hot-plug slots).
2. After a system reboot (if necessary) test software examines the PCI Express Capabilities register for each Downstream Port in the system.

3. For each Downstream Port that has a PCI Express Capabilities register that has both the Slot Implemented field (PCI Express Capabilities register) returns 1 and the Hot-Plug Capable field (Slot Capabilities register) returns 1:
 - a. Test software identifies for the user, the slot currently under test by doing the following:
 - i. A message is presented displaying to the user, the value read from the Physical Slot Number field (Slot Capabilities register).
 - ii. If the Attention Indicator Present field (Slot Capabilities register) returns 1, test software writes the Attention Indicator Control field (Slot Control register) with 10b (Blink). If the No Command Completed Support field (Slot Capabilities register) is 1, test software polls the Command Completed field (Slot Status register) until it reads back a 1. If this does not occur after 1 second of polling, then the write to the Slot Control register is repeated. If this still does not occur after 10 repeats, the remainder of this step is skipped and the test continues with the next step. When the Command Completed field (Slot Status register) returns 1, test software writes the Command Completed field (Slot Status register) with 1 (to clear the status).
 - b. Test software blinks the power indicator for the slot currently under test by doing the following:
 - i. If the Power Indicator Present field (Slot Capabilities register) returns 1, test software writes the Power Indicator Control field (Slot Control register) with 10b (Blink). If the No Command Completed Support field (Slot Capabilities register) is 1, test software polls the Command Completed field (Slot Status register) until it reads back a 1. If this does not occur after 1 second of polling, then the write to the Slot Control register is repeated. If this still does not occur after 10 repeats, the remainder of this step is skipped and the test continues with the next step. When the Command Completed field (Slot Status register) returns 1, test software writes the Command Completed field (Slot Status register) with 1 (to clear the status). Test software reads the Power Indicator Control field of the Slot Control register to verify that it returns 10b (Blink) and if it does not the test fails for this slot, and the remaining steps are skipped.
 - c. Test software powers down the slot currently under test by doing the following:
 - i. If the Power Controller Present field (Slot Capabilities register) returns 1, test software writes the Power Controller Control field (Slot Control register) with 1 (Power Off). If the No Command Completed Support field (Slot Capabilities register) is 1, test software polls the Command Completed field (Slot Status register) until it reads back a 1. If this does not occur after 1 second of polling, then the write to the Slot Control register is repeated. If this still does not occur after 10 repeats, the remainder of this step is skipped and the test continues with the next step. When the Command Completed field (Slot Status register) returns 1, test software writes the Command Completed field (Slot Status register) with 1 (to clear the status). Test software reads the Power Controller Control field of the Slot Control register to verify that it returns 1 (slot power is off) and if it does not the test fails for this slot, and the remaining steps are skipped.
 - ii. Test software waits for 1 second to allow power to settle in the off state.

- d. Test software turns off the power indicator for the slot currently under test by doing the following:
 - i. If the Power Indicator Present field (Slot Capabilities register) returns 1, test software writes the Power Indicator Control field (Slot Control register) with 11b (Off). If the No Command Completed Support field (Slot Capabilities register) is 1, test software polls the Command Completed field (Slot Status register) until it reads back a 1. If this does not occur after 1 second of polling, then the write to the Slot Control register is repeated. If this still does not occur after 10 repeats, the remainder of this step is skipped and the test continues with the next step. When the Command Completed field (Slot Status register) returns 1, test software writes the Command Completed field (Slot Status register) with 1 (to clear the status). Test software reads the Power Indicator Control field of the Slot Control register to verify that it returns 11b (Off) and if it does not the test fails for this slot, and the remaining steps are skipped.
- e. Test software opens the slot currently under test by doing the following:
 - i. If the Electromechanical Interlock Present field (Slot Capabilities register) returns 1, test software reads the Electromechanical Interlock Status field (Slot Status register) and only if it returns 1 (engaged) does it write the Electromechanical Interlock Control field (Slot Control register) with 1 (toggle interlock). If the No Command Completed Support field (Slot Capabilities register) is 1, test software polls the Command Completed field (Slot Status register) until it reads back a 1. If this does not occur after 1 second of polling, then the write to the Slot Control register is repeated. If this still does not occur after 10 repeats, the remainder of this step is skipped and the test continues with the next step. When the Command Completed field (Slot Status register) returns 1, test software writes the Command Completed field (Slot Status register) with 1 (to clear the status).
- f. Test software checks the attention button for the slot currently under test by doing the following:
 - i. If the Attention Button Present field (Slot Capabilities register) returns 1, test software writes the Attention Button Pressed field (Slot Control register) with 1 to clear the previous status. Test software then prompts the user to press the attention button for the slot under test. Test software then checks that the Attention Button Pressed field (Slot Status register) returns 1 and if it does not the test fails for this slot and the test continues with the next step. Test software writes the Attention Button Pressed field (Slot Status register) with 1 (to clear the status).
- g. If the Data Link Layer Link Active Reporting Capable field (Link Capabilities register) returns 1, test software writes the Data Link Layer State Changed field (Slot Status register) with 1 (to clear the status).
- h. Test software writes the Presence Detect Changed field (Slot Status register) with 1 (to clear the status).
- i. Test software prompts the user to insert a hot-pluggable PCI Express card into the slot under test, but to not yet close any mechanical retention mechanism (if present).

- j. Test software locks the slot currently under test by doing the following:
 - i. If the MRL Sensor Present field (Slot Capabilities register) returns 1, test software writes the MRL Sensor Changed field (Slot Status register) with 1 (to clear the status).
 - ii. If the MRL Sensor Present field (Slot Capabilities register) returns 1, test software presents a message to the user, asking that they move the mechanical release latch for the slot under test to the closed position.
 - iii. If the MRL Sensor Present field (Slot Capabilities register) returns 1, test software checks that the MRL Sensor Changed field (Slot Status register) returns 1, and if it does not the test fails for this slot and the test continues with the next step. Test software writes the MRL Sensor Changed field (Slot Status register) with 1 (to clear the status).
 - iv. If the MRL Sensor Present field (Slot Capabilities register) returns 1, test software checks that the MRL Sensor State field (Slot Status register) returns 0 (Closed), and if it does not the test fails for this slot the remaining steps are skipped.
- k. Test software closes the slot currently under test by doing the following:
 - i. If the Electromechanical Interlock Present field (Slot Capabilities register) returns 1, test software reads the Electromechanical Interlock Status field (Slot Status register) and only if it returns 0 (disengaged) does it write the Electromechanical Interlock Control field (Slot Control register) with 1 (toggle interlock). If the No Command Completed Support field (Slot Capabilities register) is 1, test software polls the Command Completed field (Slot Status register) until it reads back a 1. If this does not occur after 1 second of polling, then the write to the Slot Control register is repeated. If this still does not occur after 10 repeats, the remainder of this step is skipped and the test continues with the next step. When the Command Completed field (Slot Status register) returns 1, test software writes the Command Completed field (Slot Status register) with 1 (to clear the status). Test software reads the Electromechanical Interlock Status field of the Slot Status register to verify that it returns 1 (engaged) and if it does not the test fails for this slot, and the remaining steps are skipped.
- l. Test software powers up the slot currently under test by doing the following:
 - i. If the Power Controller Present field (Slot Capabilities register) returns 1, test software writes the Power Fault Detected field (Slot Status register) with 1 (to clear the status).
 - ii. If the Power Controller Present field (Slot Capabilities register) returns 1, test software writes the Power Controller Control field (Slot Control register) with 0 (Power On). If the No Command Completed Support field (Slot Capabilities register) is 1, test software polls the Command Completed field (Slot Status register) until it reads back a 1. If this does not occur after 1 second of polling, then the write to the Slot Control register is repeated. If this still does not occur after 10 repeats, then the test fails for this slot, and the remaining steps are skipped. When the Command Completed field (Slot Status register) returns 1, test software writes the Command Completed field (Slot Status register) with 1 (to clear the status). Test software reads the Power Controller Control field of the Slot Control register to verify that it returns 0 (slot power is on) and if it does not the test fails for this slot, and the remaining steps are skipped.
- m. Test software check for power faults in the slot currently under test by doing the following:
 - i. If the Power Controller Present field (Slot Capabilities register) returns 1, test software reads the Power Fault Detected field of the Slot Status register to verify that it returns 0 (no faults) and if it does not the test fails for this slot, both step c) and step d) are repeated (to power down the slot and turn off the power indicator), and the remaining steps are skipped.

- n. Test software turns on the power indicator for the slot currently under test by doing the following:
 - i. If the Power Indicator Present field (Slot Capabilities register) returns 1, test software writes the Power Indicator Control field (Slot Control register) with 01b (On). If the No Command Completed Support field (Slot Capabilities register) is 1, test software polls the Command Completed field (Slot Status register) until it reads back a 1. If this does not occur after 1 second of polling, then the write to the Slot Control register is repeated. If this still does not occur after 10 repeats, the remainder of this step is skipped and the test continues with the next step. When the Command Completed field (Slot Status register) returns 1, test software writes the Command Completed field (Slot Status register) with 1 (to clear the status). Test software reads the Power Indicator Control field of the Slot Control register to verify that it returns 01b (On) and if it does not the test fails for this slot, and the remaining steps are skipped.
 - o. Test software checks that the Presence Detect Changed field of the Slot Status register returns 1 (presence detect state has changed) and if it does not the test fails for this slot and the remaining steps are skipped. Test software writes the Presence Detect Changed field (Slot Status register) with 1 (to clear the status).
 - p. Test software checks that the Presence Detect State field of the Slot Status register returns 1 (card present) and if it does not the test fails for this slot and the remaining steps are skipped.
 - q. Test software waits for 1 second to allow Data Link Layer to become active. (Note: software is required to allow 1 second from the hot-plug event until it determines the device does not function.)
 - r. Test software reads that the Link Disable field of the Link Control register and only if it returns 1 (disabled), it writes the Link Disable field (Link Control register) with a 0 (to re-enable the link) using a WORD access, while preserving all the other fields in this register.
 - s. Test software then repeatedly reads the Link Training field of the Link Status register, until it returns 0. If it does not return 0 within 1 second the test fails for this slot and the remaining steps are skipped.
 - t. If the Data Link Layer Link Active Reporting Capable field (Link Capabilities register) returns 1, test software checks that the Data Link Layer State Changed field of the Slot Status register returns 1 (data link layer link active state has changed) and if it does not the test fails for this slot and the remaining steps are skipped. Test software writes the Data Link Layer State Changed field (Slot Status register) with 1 (to clear the status).
 - u. Test software configures at least one of the functions on the PCI Express card added to the slot under testing using the standard method described in Section 2.1.1.2. If it does not configure properly the test fails for this slot and the remaining steps are skipped.
 - v. Test software clears the attention indicator for the slot currently under test by doing the following:
 - i. If the Attention Indicator Present field (Slot Capabilities register) returns 1, test software writes the Attention Indicator Control field (Slot Control register) with 11b (Off). If the No Command Completed Support field (Slot Capabilities register) is 1, test software polls the Command Completed field (Slot Status register) until it reads back a 1. If this does not occur after 1 second of polling, then the write to the Slot Control register is repeated. If this still does not occur after 10 repeats, the remainder of this step is skipped and the test continues with the next step. When the Command Completed field (Slot Status register) returns 1, test software writes the Command Completed field (Slot Status register) with 1 (to clear the status).

4. The test is repeated with all PCI Express cards that are supported by the slot under test's form factor (e.g., different add-in card connector widths, different add-in card link speeds, different add-in card power consumptions, etc.).

The test *fails* if:

- ☐ The Power Controller Control field does not properly reflect the last command (power on/power off) written to it, for a slot that indicates it implements a power controller.
- ☐ Writing a 1 to the Power Controller Control field does not turn power off to a slot that indicates it implements a power controller.
- ☐ The Power Indicator Control field does not properly reflect the last command (power indicator on/power indicator off) written to it, for a slot that indicates it implements a power indicator.
- ☐ Pressing the attention button does not cause the Attention Button Pressed field to be cleared, for a slot that indicates it implements an attention button.
- ☐ Closing the mechanical retention latch on a slot does not cause the MRL Sensor Changed field to be set, for a slot that indicates it implements a MRL sensor.
- ☐ Closing the mechanical retention latch on a slot does not cause the MRL Sensor State field to be cleared, for a slot that indicates it implements a MRL sensor.
- ☐ The Electromechanical Interlock field does not indicate the engaged state when the interlock is toggled to the engaged setting, for a slot that indicates it implements an electromechanical interlock.
- ☐ The hot-plug command to power on the slot does not complete properly, for a slot that indicates it implements a power controller.
- ☐ Writing a 0 to the Power Controller Control field does not turn power on to a slot that indicates it implements a power controller.
- ☐ The Power Fault Detected field indicates a power fault following a power on of the slot, for a slot that indicates it implements a power controller.
- ☐ Powering on a slot with a card installed does not cause the Presence Detect Changed field to be set.
- ☐ Powering on a slot with a card installed does not cause the Presence Detect State field to be set.
- ☐ Powering on a slot with a card installed does not result in successful link training.
- ☐ Powering on a slot with a card installed does not cause the Data Link Layer State Changed field to be set, for a slot that indicates it implements data link layer active reporting capability.
- ☐ A card that has been hot inserted cannot be successfully configured.

B.9 TD_3_5 Basic Hot-Plug Removal

Test TD_3_5 is no longer supported as it requires the removal of all PCI Express cards, which cannot be done if some cards are required to support the test itself (such as a display card).

The test verifies that the system under test can properly handle hot-plug situations of hot remove and that each downstream port connected to a physical slot accurately reports its hot-plug state in its Slot Capabilities, Slot Control, and Slot Status registers. (The test prompts the user to populate/unpopulate the hot-plug slots under test. If an Attention Button is implemented, the test also prompts the user to press it.)

Relevant Specifications

❑ *PCI Express Base Specification*

(See Section 1.3 for additional specification revision details.)

Applicable Device/Port Types

This test is run on downstream ports of Root Ports, Switch Downstream Ports, and PCI/PCI-X to PCI Express Bridges.

Starting Configuration

For a non-Root Port function under test, it is completely uninitialized following a reset, except that the bus number fields are initialized with valid numbers.

The immediate downstream device (the upstream port of the link connected to the function under test) is completely uninitialized following a reset.

Overview of Test Steps

Test software performs the following steps:

1. Prompt the user to power down (if necessary) and install PCI Express cards in all physically exposed slots (hot-plug slots and optionally non-hot-plug slots).
2. After a system reboot (if necessary) test software examines the PCI Express Capabilities register for each Downstream Port in the system.
3. For each Downstream Port that has a PCI Express Capabilities register that has both the Slot Implemented field (PCI Express Capabilities register) returns 1 and the Hot-Plug Capable field (Slot Capabilities register) returns 1:
 - a. Test software identifies for the user, the slot currently under test by doing the following:
 - i. A message is presented displaying to the user, the value read from the Physical Slot Number field (Slot Capabilities register).
 - ii. If the Attention Indicator Present field (Slot Capabilities register) returns 1, test software writes the Attention Indicator Control field (Slot Control register) with 10b (Blink). If the No Command Completed Support field (Slot Capabilities register) is 1, test software polls the Command Completed field (Slot Status register) until it reads back a 1. If this does not occur after 1 second of polling, then the write to the Slot Control register is repeated. If this still does not occur after 10 repeats, the remainder of this step is skipped and the test continues with the next step. When the Command Completed field (Slot Status register) returns 1, test software writes the Command Completed field (Slot Status register) with 1 (to clear the status).

- b. Test software checks the attention button for the slot currently under test by doing the following:
 - i. If the Attention Button Present field (Slot Capabilities register) returns 1, test software writes the Attention Button Pressed field (Slot Control register) with 1 to clear the previous status. Test software then prompts the user to press the attention button for the slot under test. Test software then checks that the Attention Button Pressed field (Slot Status register) returns 1 and if it does not the test fails for this slot and the test continues with the next step. Test software writes the Attention Button Pressed field (Slot Status register) with 1 (to clear the status).
- c. Test software writes the Link Disable field (Link Control register) with a 1 (to disable the link) using a WORD access, while preserving all the other fields in this register.
- d. Test software blinks the power indicator for the slot currently under test by doing the following:
 - i. If the Power Indicator Present field (Slot Capabilities register) returns 1, test software writes the Power Indicator Control field (Slot Control register) with 10b (Blink). If the No Command Completed Support field (Slot Capabilities register) is 1, test software polls the Command Completed field (Slot Status register) until it reads back a 1. If this does not occur after 1 second of polling, then the write to the Slot Control register is repeated. If this still does not occur after 10 repeats, the remainder of this step is skipped and the test continues with the next step. When the Command Completed field (Slot Status register) returns 1, test software writes the Command Completed field (Slot Status register) with 1 (to clear the status). Test software reads the Power Indicator Control field of the Slot Control register to verify that it returns 10b (Blink) and if it does not the test fails for this slot, and the remaining steps are skipped.
- e. If the Data Link Layer Link Active Reporting Capable field (Link Capabilities register) returns 1, test software writes the Data Link Layer State Changed field (Slot Status register) with 1 (to clear the status).
- f. Test software writes the Presence Detect Changed field (Slot Status register) with 1 (to clear the status).
- g. Test software powers down the slot currently under test by doing the following:
 - i. If the Power Controller Present field (Slot Capabilities register) returns 1, test software writes the Power Fault Detected field (Slot Status register) with 1 (to clear the status).
 - ii. If the Power Controller Present field (Slot Capabilities register) returns 1, test software writes the Power Controller Control field (Slot Control register) with 1 (Power Off). If the No Command Completed Support field (Slot Capabilities register) is 1, test software polls the Command Completed field (Slot Status register) until it reads back a 1. If this does not occur after 1 second of polling, then the write to the Slot Control register is repeated. If this still does not occur after 10 repeats, the remainder of this step is skipped and the test continues with the next step. When the Command Completed field (Slot Status register) returns 1, test software writes the Command Completed field (Slot Status register) with 1 (to clear the status). Test software reads the Power Controller Control field of the Slot Control register to verify that it returns 1 (slot power is off) and if it does not the test fails for this slot, and the remaining steps are skipped.
 - iii. Test software waits for 1 second to allow power to settle in the off state.
- h. Test software turns off the power indicator for the slot currently under test by doing the following:
 - i. If the Power Indicator Present field (Slot Capabilities register) returns 1, test software writes the Power Indicator Control field (Slot Control register) with 11b (Off). If the

- No Command Completed Support field (Slot Capabilities register) is 1, test software polls the Command Completed field (Slot Status register) until it reads back a 1. If this does not occur after 1 second of polling, then the write to the Slot Control register is repeated. If this still does not occur after 10 repeats, the remainder of this step is skipped and the test continues with the next step. When the Command Completed field (Slot Status register) returns 1, test software writes the Command Completed field (Slot Status register) with 1 (to clear the status). Test software reads the Power Indicator Control field of the Slot Control register to verify that it returns 11b (Off) and if it does not the test fails for this slot, and the remaining steps are skipped.
- i. Test software opens the slot currently under test by doing the following:
 - i. If the Electromechanical Interlock Present field (Slot Capabilities register) returns 1, test software reads the Electromechanical Interlock Status field (Slot Status register) and only if it returns 1 (engaged) does it write the Electromechanical Interlock Control field (Slot Control register) with 1 (toggle interlock). If the No Command Completed Support field (Slot Capabilities register) is 1, test software polls the Command Completed field (Slot Status register) until it reads back a 1. If this does not occur after 1 second of polling, then the write to the Slot Control register is repeated. If this still does not occur after 10 repeats, the remainder of this step is skipped and the test continues with the next step. When the Command Completed field (Slot Status register) returns 1, test software writes the Command Completed field (Slot Status register) with 1 (to clear the status). Test software reads the Electromechanical Interlock Status field of the Slot Status register to verify that it returns 0 (disengaged) and if it does not the test fails for this slot, and the remaining steps are skipped.
 - j. Test software unlocks the slot currently under test by doing the following:
 - i. If the MRL Sensor Present field (Slot Capabilities register) returns 1, test software writes the MRL Sensor Changed field (Slot Status register) with 1 (to clear the status).
 - ii. If the MRL Sensor Present field (Slot Capabilities register) returns 1, test software presents a message to the user, asking that they move the mechanical release latch for the slot under test to the open position.
 - iii. If the MRL Sensor Present field (Slot Capabilities register) returns 1, test software checks that the MRL Sensor Changed field (Slot Status register) returns 1, and if it does not the test fails for this slot and the test continues with the next step. Test software writes the MRL Sensor Changed field (Slot Status register) with 1 (to clear the status).
 - iv. If the MRL Sensor Present field (Slot Capabilities register) returns 1, test software checks that the MRL Sensor State field (Slot Status register) returns 1 (Open), and if it does not the test fails for this slot and the test continues with the next step.
 - k. Test software prompts the user to remove the hot-pluggable PCI Express card from the slot under test.
 - l. Test software check for power faults in the slot currently under test by doing the following:
 - i. If the Power Controller Present field (Slot Capabilities register) returns 1, test software reads the Power Fault Detected field of the Slot Status register to verify that it returns 0 (no faults) and if it does not the test fails for this slot, both step g) and step h) are repeated (to power down the slot and turn off the power indicator), and the remaining steps are skipped.
 - m. Test software checks that the Presence Detect Changed field of the Slot Status register returns 1 (presence detect state has changed) and if it does not the test fails for this slot and the remaining steps are skipped. Test software writes the Presence Detect Changed field (Slot Status register) with 1 (to clear the status).

- n. Test software checks that the Presence Detect State field of the Slot Status register returns 0 (slot empty) and if it does not the test fails for this slot and the remaining steps are skipped.
 - o. If the Data Link Layer Link Active Reporting Capable field (Link Capabilities register) returns 1, test software checks that the Data Link Layer State Changed field of the Slot Status register returns 1 (data link layer link active state has changed) and if it does not the test fails for this slot and the remaining steps are skipped. Test software writes the Data Link Layer State Changed field (Slot Status register) with 1 (to clear the status).
 - p. Test software clears the attention indicator for the slot currently under test by doing the following:
 - i. If the Attention Indicator Present field (Slot Capabilities register) returns 1, test software writes the Attention Indicator Control field (Slot Control register) with 11b (Off). If the No Command Completed Support field (Slot Capabilities register) is 1, test software polls the Command Completed field (Slot Status register) until it reads back a 1. If this does not occur after 1 second of polling, then the write to the Slot Control register is repeated. If this still does not occur after 10 repeats, the remainder of this step is skipped and the test continues with the next step. When the Command Completed field (Slot Status register) returns 1, test software writes the Command Completed field (Slot Status register) with 1 (to clear the status).
4. The test is repeated with all PCI Express cards that are supported by the slot under test's form factor (e.g., different add-in card connector widths, different add-in card link speeds, different add-in card power consumptions, etc.).

The test *fails* if:

- ☐ Pressing the attention button does not cause the Attention Button Pressed field to be cleared, for a slot that indicates it implements an attention button.
- ☐ The Power Controller Control field does not properly reflect the last command (power on/power off) written to it, for a slot that indicates it implements a power controller.
- ☐ The hot-plug command to power off the slot does not complete properly, for a slot that indicates it implements a power controller.
- ☐ Writing a 1 to the Power Controller Control field does not turn power off to a slot that indicates it implements a power controller.
- ☐ The Power Indicator Control field does not properly reflect the last command (power indicator on/power indicator off) written to it, for a slot that indicates it implements a power indicator.
- ☐ The Electromechanical Interlock field does not indicate the engaged state when the interlock is toggled to the engaged setting, for a slot that indicates it implements an electromechanical interlock.
- ☐ Opening the mechanical retention latch on a slot does not cause the MRL Sensor Changed field to be set, for a slot that indicates it implements a MRL sensor.
- ☐ Opening the mechanical retention latch on a slot does not cause the MRL Sensor State field to be set, for a slot that indicates it implements a MRL sensor.
- ☐ Removing a card from a slot does not cause the Presence Detect Changed field to be set.
- ☐ Removing a card from a slot does not cause the Presence Detect State field to be cleared.
- ☐ Removing a card from a slot does not cause the Data Link Layer State Changed field to be set, for a slot that indicates it implements data link layer active reporting capability.

B.10TD_3_6 Basic Hot-Plug Surprise Removal

Test TD_3_6 is no longer supported as it requires the removal of all PCI Express cards, which cannot be done if some cards are required to support the test itself (such as a display card).

The test verifies that the system under test can properly handle hot-plug situations of surprise hot remove and that each downstream port connected to a physical slot accurately reports its hot-plug state in its Slot Capabilities, Slot Control, and Slot Status registers. (The test prompts the user to populate/unpopulate the hot-plug slots under test.)

Relevant Specifications

❑ *PCI Express Base Specification*

(See Section 1.3 for additional specification revision details.)

Applicable Device/Port Types

This test is run on downstream ports of Root Ports, Switch Downstream Ports, and PCI/PCI-X to PCI Express Bridges.

Starting Configuration

For a non-Root Port function under test, it is completely uninitialized following a reset, except that the bus number fields are initialized with valid numbers.

The immediate downstream device (the upstream port of the link connected to the function under test) is completely uninitialized following a reset.

Overview of Test Steps

Test software performs the following steps:

1. Prompt the user to power down (if necessary) and install PCI Express cards in all physically exposed slots (hot-plug slots and optionally non-hot-plug slots).
2. After a system reboot (if necessary) test software examines the PCI Express Capabilities register for each Downstream Port in the system.
3. For each Downstream Port that has a PCI Express Capabilities register that has all three of the Slot Implemented field (PCI Express Capabilities register) returns 1, the Hot-Plug Capable field (Slot Capabilities register) returns 1, and the Hot-Plug Surprise field (Slot Capabilities register) returns 1:
 - a. Test software identifies for the user, the slot currently under test by doing the following:
 - i. A message is presented displaying to the user, the value read from the Physical Slot Number field (Slot Capabilities register).
 - ii. If the Attention Indicator Present field (Slot Capabilities register) returns 1, test software writes the Attention Indicator Control field (Slot Control register) with 10b (Blink). If the No Command Completed Support field (Slot Capabilities register) is 1, test software polls the Command Completed field (Slot Status register) until it reads back a 1. If this does not occur after 1 second of polling, then the write to the Slot Control register is repeated. If this still does not occur after 10 repeats, the remainder of this step is skipped and the test continues with the next step. When the Command Completed field (Slot Status register) returns 1, test software writes the Command Completed field (Slot Status register) with 1 (to clear the status).

- b. If the Attention Button Present field (Slot Capabilities register) returns 1, test software writes the Attention Button Pressed field (Slot Control register) with 1 (to clear the status).
 - c. If the Data Link Layer Link Active Reporting Capable field (Link Capabilities register) returns 1, test software writes the Data Link Layer State Changed field (Slot Status register) with 1 (to clear the status).
 - d. Test software writes the Presence Detect Changed field (Slot Status register) with 1 (to clear the status).
 - e. Test software checks the attention button for the slot currently under test by doing the following:
 - i. If the Attention Button Present field (Slot Capabilities register) returns 1, test software checks that the Attention Button Pressed field (Slot Status register) returns 0 and if it does not the test fails for this slot and the test continues with the next step.
 - f. Test software prompts the user to remove the hot-pluggable PCI Express card from the slot under test.
 - g. Test software checks that the Presence Detect Changed field of the Slot Status register returns 1 (presence detect state has changed) and if it does not the test fails for this slot and the remaining steps are skipped. Test software writes the Presence Detect Changed field (Slot Status register) with 1 (to clear the status).
 - h. Test software checks that the Presence Detect State field of the Slot Status register returns 0 (slot empty) and if it does not the test fails for this slot and the remaining steps are skipped.
 - i. If the Data Link Layer Link Active Reporting Capable field (Link Capabilities register) returns 1, test software checks that the Data Link Layer State Changed field of the Slot Status register returns 1 (data link layer link active state has changed) and if it does not the test fails for this slot and the remaining steps are skipped. Test software writes the Data Link Layer State Changed field (Slot Status register) with 1 (to clear the status).
 - j. Test software clears the attention indicator for the slot currently under test by doing the following:
 - i. If the Attention Indicator Present field (Slot Capabilities register) returns 1, test software writes the Attention Indicator Control field (Slot Control register) with 11b (Off). If the No Command Completed Support field (Slot Capabilities register) is 1, test software polls the Command Completed field (Slot Status register) until it reads back a 1. If this does not occur after 1 second of polling, then the write to the Slot Control register is repeated. If this still does not occur after 10 repeats, the remainder of this step is skipped and the test continues with the next step. When the Command Completed field (Slot Status register) returns 1, test software writes the Command Completed field (Slot Status register) with 1 (to clear the status).
4. The test is repeated with all PCI Express cards that are supported by the slot under test's form factor (e.g., different add-in card connector widths, different add-in card link speeds, different add-in card power consumptions, etc.).

The test *fails* if:

- ☐ The Attention Button Pressed field does not stay cleared, for a slot that indicates it implements an attention button.
- ☐ Surprise removing a card from a slot does not cause the Presence Detect Changed field to be set.
- ☐ Surprise removing a card from a slot does not cause the Presence Detect State field to be cleared.
- ☐ Surprise removing a card from a slot does not cause the Data Link Layer State Changed field to be set, for a slot that indicates it implements data link layer active reporting capability.

B.11 TD_3_7 Attention Button MRL Indicator Control

Test TD_3_7 is no longer supported as it requires the removal of all PCI Express cards, which cannot be done if some cards are required to support the test itself (such as a display card).

The test verifies that the system under test can properly reports attention buttons, MRL sensors, power indicators, and attention indicators where they are implemented, using its Slot Capabilities registers. (The test prompts the user to visually check for the presence of attention buttons, MRL sensors, power indicators, and attention indicators. If an Attention Button is implemented, the test also prompts the user to press it. If a mechanical retention latch is present, the test also prompts the user to toggle it.)

Relevant Specifications

- ☐ *PCI Express Base Specification*

(See Section 1.3 for additional specification revision details.)

Applicable Device/Port Types

This test is run on downstream ports of Root Ports, Switch Downstream Ports, and PCI/PCI-X to PCI Express Bridges.

Starting Configuration

For a non-Root Port function under test, it is completely uninitialized following a reset, except that the bus number fields are initialized with valid numbers.

The immediate downstream device (the upstream port of the link connected to the function under test) is completely uninitialized following a reset.

Overview of Test Steps

Test software performs the following steps:

1. Test software checks each Downstream Port and records all those that have a PCI Express Capabilities register that has the Slot Implemented field (PCI Express Capabilities register) return 1.
2. Test software writes the Slot Control register, of each Downstream Port that has a PCI Express Capabilities register that has the Slot Implemented field (PCI Express Capabilities register) return 1, with a value that sets to 00b (Reserved) the following fields: Attention Indicator Control; Power Indicator Control.

3. Test software writes the Slot Status register, of each Downstream Port that has a PCI Express Capabilities register that has the Slot Implemented field (PCI Express Capabilities register) return 1, with a value that sets to 1 (to clear the status) the following fields: Attention Button Pressed; MRL Sensor Changed.
4. For each Downstream Port that has a PCI Express Capabilities register that has the Slot Implemented field (PCI Express Capabilities register) return 1:
 - a. Test software identifies for the user, the slot currently under test by doing the following:
 - i. A message is presented displaying to the user, the value read from the Physical Slot Number field (Slot Capabilities register).
 - ii. If the Attention Indicator Present field (Slot Capabilities register) returns 1, test software writes the Attention Indicator Control field (Slot Control register) with 10b (Blink). If the No Command Completed Support field (Slot Capabilities register) is 1, test software polls the Command Completed field (Slot Status register) until it reads back a 1. If this does not occur after 1 second of polling, then the write to the Slot Control register is repeated. If this still does not occur after 10 repeats, the remainder of this step is skipped and the test continues with the next step. When the Command Completed field (Slot Status register) returns 1, test software writes the Command Completed field (Slot Status register) with 1 (to clear the status).
 - b. Test software checks the attention button for the slot currently under test by doing the following:
 - i. If the Attention Button Present field (Slot Capabilities register) returns 0, test software then prompts the user to verify that there is no attention button for the slot under test. If the user responds negatively, the test fails for this slot and the test continues with the next step.
 - ii. If the Attention Button Present field (Slot Capabilities register) returns 1, test software writes the Attention Button Pressed field (Slot Control register) with 1 to clear the previous status. Test software then prompts the user to press the attention button for the slot under test. Test software then checks that the Attention Button Pressed field (Slot Status register) returns 1 for the slot under test only and if it does not the test fails for this slot and the test continues with the next step. Test software writes the Attention Button Pressed field (Slot Status register) with 1 (to clear the status).
 - iii. Test software then checks that the Attention Button Pressed field (Slot Status register) returns 0 for each Downstream Port that has a PCI Express Capabilities register that has the Slot Implemented field (PCI Express Capabilities register) return 1, and if it does not the test fails for that individual slot and the test continues with the next step.
 - c. Test software checks the mechanical retention latch for the slot currently under test by doing the following:
 - i. If the MRL Sensor Present field (Slot Capabilities register) returns 1, test software writes the MRL Sensor Changed field (Slot Status register) with 1 (to clear the status).
 - ii. If the MRL Sensor Present field (Slot Capabilities register) returns 1, test software reads the MRL Sensor State field (Slot Status register) and records its current value.
 - iii. If the MRL Sensor Present field (Slot Capabilities register) returns 1, test software presents a message to the user, asking that they move the mechanical release latch for the slot under test to the opposite position.

- iv If the MRL Sensor Present field (Slot Capabilities register) returns 1, test software checks that the MRL Sensor Changed field (Slot Status register) returns 1, and if it does not the test fails for this slot and the test continues with the next step. Test software writes the MRL Sensor Changed field (Slot Status register) with 1 (to clear the status).
- v If the MRL Sensor Present field (Slot Capabilities register) returns 1, test software reads that the MRL Sensor State field (Slot Status register), and if it does not return the opposite value to the previously recorded value the test fails for this slot and the test continues with the next step.
- d. Test software checks the attention indicator for the slot currently under test by doing the following:
 - i If the Attention Indicator Present field (Slot Capabilities register) returns 0, test software then prompts the user to verify that there is no attention indicator for the slot under test. If the user responds negatively, the test fails for this slot and the test continues with the next step.
 - ii If the Attention Indicator Present field (Slot Capabilities register) returns 1, test software writes the Attention Indicator Control field (Slot Control register) with 01b (On). If the No Command Completed Support field (Slot Capabilities register) is 1, test software polls the Command Completed field (Slot Status register) until it reads back a 1. If this does not occur after 1 second of polling, then the write to the Slot Control register is repeated. If this still does not occur after 10 repeats, the remainder of this step is skipped and the test continues with the next step. When the Command Completed field (Slot Status register) returns 1, test software writes the Command Completed field (Slot Status register) with 1 (to clear the status). Test software checks the Attention Indicator Control field (Slot Control register) returns 01b (On), and if it does not the test fails for this slot and the test continues with the next step.
 - iii If the Attention Indicator Present field (Slot Capabilities register) returns 1, test software then prompts the user to verify that the attention indicator is fully on for the slot under test. If the user responds negatively, the test fails for this slot and the test continues with the next step.
 - iv If the Attention Indicator Present field (Slot Capabilities register) returns 1, test software writes the Attention Indicator Control field (Slot Control register) with 10b (Blink). If the No Command Completed Support field (Slot Capabilities register) is 1, test software polls the Command Completed field (Slot Status register) until it reads back a 1. If this does not occur after 1 second of polling, then the write to the Slot Control register is repeated. If this still does not occur after 10 repeats, the remainder of this step is skipped and the test continues with the next step. When the Command Completed field (Slot Status register) returns 1, test software writes the Command Completed field (Slot Status register) with 1 (to clear the status). Test software checks the Attention Indicator Control field (Slot Control register) returns 10b (Blink), and if it does not the test fails for this slot and the test continues with the next step.
 - v If the Attention Indicator Present field (Slot Capabilities register) returns 1, test software then prompts the user to verify that the attention indicator is blinking for the slot under test. If the user responds negatively, the test fails for this slot and the test continues with the next step.

- vi If the Attention Indicator Present field (Slot Capabilities register) returns 1, test software writes the Attention Indicator Control field (Slot Control register) with 11b (Off). If the No Command Completed Support field (Slot Capabilities register) is 1, test software polls the Command Completed field (Slot Status register) until it reads back a 1. If this does not occur after 1 second of polling, then the write to the Slot Control register is repeated. If this still does not occur after 10 repeats, the remainder of this step is skipped and the test continues with the next step. When the Command Completed field (Slot Status register) returns 1, test software writes the Command Completed field (Slot Status register) with 1 (to clear the status). Test software checks the Attention Indicator Control field (Slot Control register) returns 11b (Off), and if it does not the test fails for this slot and the test continues with the next step.
- vii If the Attention Indicator Present field (Slot Capabilities register) returns 1, test software then prompts the user to verify that the attention indicator is fully off for the slot under test. If the user responds negatively, the test fails for this slot and the test continues with the next step.
- viii If the Attention Indicator Present field (Slot Capabilities register) returns 1, test software writes the Attention Indicator Control field (Slot Control register) with 01b (On). If the No Command Completed Support field (Slot Capabilities register) is 1, test software polls the Command Completed field (Slot Status register) until it reads back a 1. If this does not occur after 1 second of polling, then the write to the Slot Control register is repeated. If this still does not occur after 10 repeats, the remainder of this step is skipped and the test continues with the next step. When the Command Completed field (Slot Status register) returns 1, test software writes the Command Completed field (Slot Status register) with 1 (to clear the status). Test software checks the Attention Indicator Control field (Slot Control register) returns 01b (On), and if it does not the test fails for this slot and the test continues with the next step.
- ix Test software suspends the system to S3 and then resumes it again.
- x If the Attention Indicator Present field (Slot Capabilities register) returns 1, test software then checks the Attention Indicator Control field (Slot Control register) returns 01b (On), and if it does not the test fails for this slot and the test continues with the next step.
- xi If the Attention Indicator Present field (Slot Capabilities register) returns 1, test software then prompts the user to verify that the attention indicator is still fully on for the slot under test. If the user responds negatively, the test fails for this slot and the test continues with the next step.
- xii If the Attention Indicator Present field (Slot Capabilities register) returns 1, test software writes the Attention Indicator Control field (Slot Control register) with 10b (Blink). If the No Command Completed Support field (Slot Capabilities register) is 1, test software polls the Command Completed field (Slot Status register) until it reads back a 1. If this does not occur after 1 second of polling, then the write to the Slot Control register is repeated. If this still does not occur after 10 repeats, the remainder of this step is skipped and the test continues with the next step. When the Command Completed field (Slot Status register) returns 1, test software writes the Command Completed field (Slot Status register) with 1 (to clear the status). Test software checks the Attention Indicator Control field (Slot Control register) returns 10b (Blink), and if it does not the test fails for this slot and the test continues with the next step.

- xiii Test software suspends the system to S3 and then resumes it again.
- xiv If the Attention Indicator Present field (Slot Capabilities register) returns 1, test software then checks the Attention Indicator Control field (Slot Control register) returns 10b (Blink), and if it does not the test fails for this slot and the test continues with the next step.
- xv If the Attention Indicator Present field (Slot Capabilities register) returns 1, test software then prompts the user to verify that the attention indicator is still blinking for the slot under test. If the user responds negatively, the test fails for this slot and the test continues with the next step.
- xvi If the Attention Indicator Present field (Slot Capabilities register) returns 1, test software writes the Attention Indicator Control field (Slot Control register) with 11b (Off). If the No Command Completed Support field (Slot Capabilities register) is 1, test software polls the Command Completed field (Slot Status register) until it reads back a 1. If this does not occur after 1 second of polling, then the write to the Slot Control register is repeated. If this still does not occur after 10 repeats, the remainder of this step is skipped and the test continues with the next step. When the Command Completed field (Slot Status register) returns 1, test software writes the Command Completed field (Slot Status register) with 1 (to clear the status).
- e. Test software checks the power indicator for the slot currently under test by doing the following:
 - i If the Power Indicator Present field (Slot Capabilities register) returns 0, test software then prompts the user to verify that there is no power indicator for the slot under test. If the user responds negatively, the test fails for this slot and the test continues with the next step.
 - ii If the Power Indicator Present field (Slot Capabilities register) returns 1, test software writes the Power Indicator Control field (Slot Control register) with 01b (On). If the No Command Completed Support field (Slot Capabilities register) is 1, test software polls the Command Completed field (Slot Status register) until it reads back a 1. If this does not occur after 1 second of polling, then the write to the Slot Control register is repeated. If this still does not occur after 10 repeats, the remainder of this step is skipped and the test continues with the next step. When the Command Completed field (Slot Status register) returns 1, test software writes the Command Completed field (Slot Status register) with 1 (to clear the status). Test software checks the Power Indicator Control field (Slot Control register) returns 01b (On), and if it does not the test fails for this slot and the test continues with the next step.
 - iii If the Power Indicator Present field (Slot Capabilities register) returns 1, test software then prompts the user to verify that the power indicator is fully on for the slot under test. If the user responds negatively, the test fails for this slot and the test continues with the next step.

- iv If the Power Indicator Present field (Slot Capabilities register) returns 1, test software writes the Power Indicator Control field (Slot Control register) with 10b (Blink). If the No Command Completed Support field (Slot Capabilities register) is 1, test software polls the Command Completed field (Slot Status register) until it reads back a 1. If this does not occur after 1 second of polling, then the write to the Slot Control register is repeated. If this still does not occur after 10 repeats, the remainder of this step is skipped and the test continues with the next step. When the Command Completed field (Slot Status register) returns 1, test software writes the Command Completed field (Slot Status register) with 1 (to clear the status). Test software checks the Power Indicator Control field (Slot Control register) returns 10b (Blink), and if it does not the test fails for this slot and the test continues with the next step.
- v If the Power Indicator Present field (Slot Capabilities register) returns 1, test software then prompts the user to verify that the Power indicator is blinking for the slot under test. If the user responds negatively, the test fails for this slot and the test continues with the next step.
- vi If the Power Indicator Present field (Slot Capabilities register) returns 1, test software writes the Power Indicator Control field (Slot Control register) with 11b (Off). If the No Command Completed Support field (Slot Capabilities register) is 1, test software polls the Command Completed field (Slot Status register) until it reads back a 1. If this does not occur after 1 second of polling, then the write to the Slot Control register is repeated. If this still does not occur after 10 repeats, the remainder of this step is skipped and the test continues with the next step. When the Command Completed field (Slot Status register) returns 1, test software writes the Command Completed field (Slot Status register) with 1 (to clear the status). Test software checks the Power Indicator Control field (Slot Control register) returns 11b (Off), and if it does not the test fails for this slot and the test continues with the next step.
- vii If the Power Indicator Present field (Slot Capabilities register) returns 1, test software then prompts the user to verify that the power indicator is fully off for the slot under test. If the user responds negatively, the test fails for this slot and the test continues with the next step.
- viii If the Power Indicator Present field (Slot Capabilities register) returns 1, test software writes the Power Indicator Control field (Slot Control register) with 01b (On). If the No Command Completed Support field (Slot Capabilities register) is 1, test software polls the Command Completed field (Slot Status register) until it reads back a 1. If this does not occur after 1 second of polling, then the write to the Slot Control register is repeated. If this still does not occur after 10 repeats, the remainder of this step is skipped and the test continues with the next step. When the Command Completed field (Slot Status register) returns 1, test software writes the Command Completed field (Slot Status register) with 1 (to clear the status). Test software checks the Power Indicator Control field (Slot Control register) returns 01b (On), and if it does not the test fails for this slot and the test continues with the next step.
- ix Test software suspends the system to S3 and then resumes it again.
- x If the Power Indicator Present field (Slot Capabilities register) returns 1, test software then checks the Power Indicator Control field (Slot Control register) returns 01b (On), and if it does not the test fails for this slot and the test continues with the next step.

- xi If the Power Indicator Present field (Slot Capabilities register) returns 1, test software then prompts the user to verify that the power indicator is still fully on for the slot under test. If the user responds negatively, the test fails for this slot and the test continues with the next step.
 - xii If the Power Indicator Present field (Slot Capabilities register) returns 1, test software writes the Power Indicator Control field (Slot Control register) with 10b (Blink). If the No Command Completed Support field (Slot Capabilities register) is 1, test software polls the Command Completed field (Slot Status register) until it reads back a 1. If this does not occur after 1 second of polling, then the write to the Slot Control register is repeated. If this still does not occur after 10 repeats, the remainder of this step is skipped and the test continues with the next step. When the Command Completed field (Slot Status register) returns 1, test software writes the Command Completed field (Slot Status register) with 1 (to clear the status). Test software checks the Power Indicator Control field (Slot Control register) returns 10b (Blink), and if it does not the test fails for this slot and the test continues with the next step.
 - xiii Test software suspends the system to S3 and then resumes it again.
 - xiv If the Power Indicator Present field (Slot Capabilities register) returns 1, test software then checks the Power Indicator Control field (Slot Control register) returns 10b (Blink), and if it does not the test fails for this slot and the test continues with the next step.
 - xv If the Power Indicator Present field (Slot Capabilities register) returns 1, test software then prompts the user to verify that the power indicator is still blinking for the slot under test. If the user responds negatively, the test fails for this slot and the test continues with the next step.
 - xvi If the Power Indicator Present field (Slot Capabilities register) returns 1, test software writes the Power Indicator Control field (Slot Control register) with 11b (Off). If the No Command Completed Support field (Slot Capabilities register) is 1, test software polls the Command Completed field (Slot Status register) until it reads back a 1. If this does not occur after 1 second of polling, then the write to the Slot Control register is repeated. If this still does not occur after 10 repeats, the remainder of this step is skipped and the test continues with the next step. When the Command Completed field (Slot Status register) returns 1, test software writes the Command Completed field (Slot Status register) with 1 (to clear the status).
5. The test is repeated for each slot with a PCI Express device present in the slot and without any device present in the slot (if possible).

The test *fails* if:

- ☐ A slot has a visible attention button and does not accurately report its presence.
- ☐ Pressing the attention button for a slot does not cause the Attention Button Pressed field to update.
- ☐ Changing the MRL does not cause the MRL Sensor State field to indicate the changed position and the MRL Sensor Changed field to be set.
- ☐ A slot has a visible attention indicator and does not accurately report its presence.
- ☐ An attention indicator does not correctly latch a valid value written to its control register.
- ☐ An attention indicator does not transition to the state last written to its control register.

- ❑ An attention indicator does not return to the state it was in prior to a system suspend/resume cycle.
- ❑ A slot has a visible power indicator and does not accurately report its presence.
- ❑ A power indicator does not correctly latch a valid value written to its control register.
- ❑ A power indicator does not transition to the state last written to its control register.
- ❑ A power indicator does not return to the state it was in prior to a system suspend/resume cycle.

B.12TD_3_8 Link Retraining Stress

Test TD_3_8 is no longer supported as it requires system to reboot while the test is running, which the test cannot support.

The test verifies that the function under test implements the Retrain Link (Link Control register) behavior as defined in the relevant specifications and can properly perform link retraining when the Retrain Link bit is set. (The test prompts the user to populate the slot under test.)

Relevant Specifications

- ❑ *PCI Express Base Specification*

(See Section 1.3 for additional specification revision details.)

Applicable Device/Port Types

This test is run on downstream ports of Root Ports, Switch Downstream Ports, and PCI/PCI-X to PCI Express Bridges.

Starting Configuration

For a non-Root Port function under test, it is completely uninitialized following a reset, except that the bus number fields are initialized with valid numbers.

The immediate downstream device (the upstream port of the link connected to the function under test) is completely uninitialized following a reset.

Overview of Test Steps

Test software performs the following steps:

1. Test software checks that the function under test has a PCI Express Capabilities register that has the Slot Implemented field (PCI Express Capabilities register) return 1. If not the remaining steps are skipped (this is not a failure, but the test result is reported as skipped).
2. Prompt the user to power down (if necessary) and install a PCI Express card in the physically exposed slot under test.
3. After a system reboot (if necessary) a reset of the secondary interface is initiated by test software through writing a 1 to the Secondary Bus Reset field in the function under test (for a Bridge, the Bridge Control register in the Bridge; for a Switch, the Bridge Control register in the Switch Downstream Port) using a WORD access, while preserving all the other fields in this register, and then writing a 0 using a WORD access, while preserving all the other fields in this register. (See Section 2.1.1.1.2 for details of the reset algorithm.)
4. Configure the function under test following the procedure described in Section 2.1.1.

5. Verify that a device is present in the slot.
6. Modify some non-sticky field in the device in the slot to a non-default value. (Note: Any of the following fields can be used as long as they are implemented as RW in the target: Memory Space Enable (Command register); I/O Space Enable (Command register), Bus Master Enable (Command register), Interrupt Disable (Command register). If none of these fields are implemented as RW in the target, then other non-sticky RW fields can be used. If the test software cannot find any non-sticky RW field in the target, then this step is skipped.)
7. Test software reads the Negotiated Link Width field (Link Status register) for both the function under test and the device in the slot and records the values. If the two values do not agree, then the test fails.
8. Test software reads the Current Link Speed field (Link Status register) for both the function under test and the device in the slot and records the values. If the two values do not agree, then the test fails.
9. Test software writes a 1 to the Retrain Link field (Link Control register) of the function under test using a WORD access, while preserving all the other fields in this register.
10. The function under test is checked to see if it has completed training. Test software continuously reads the Link Training field in the Link Status register, and when it returns 0, go to the next step. If the Link Training field does not return 0 within 1 second, then the test fails, and the remaining steps are skipped.
11. Test software reads the Negotiated Link Width field (Link Status register) for both the function under test and the device in the slot and records the values. If the two values do not agree, then the test fails. If the value does not match the previously read value, then the test fails.
12. Test software reads the Current Link Speed field (Link Status register) for both the function under test and the device in the slot and records the values. If the two values do not agree, then the test fails.
13. If a non-sticky RW field was found in step 6, after the necessary delay, test software verifies the same non-sticky field (from step 6) in the device in the slot is not reset to its default value. If the value has changed to the default value, the test fails.
14. Steps 10-13 are repeated for 100 iterations.
15. The entire test is repeated for all add-in card widths that can operate in the slot under test. For example if the physical slot is x16 then x1, x4, x8, and x16 add-in cards are used (an interposer can be used to reduce the actual add-in card's width).

The test *fails* if:

- ☐ The reported negotiated link width of the function under test ever fails to match the negotiated link width of the device in the slot.
- ☐ The reported negotiated link speed of the function under test ever fails to match the negotiated link speed of the device in the slot.
- ☐ Link retraining causes the link to fail to complete link training.
- ☐ The reported negotiated link width of the function under test after link retraining fails to match the negotiated link width prior to link retraining.
- ☐ The reported negotiated link speed of the function under test after link retraining fails to match the negotiated link speed prior to link retraining.
- ☐ Link retraining causes the device in the slot to reset (restoring a non-sticky register field to its default value).

B.13 TD_3_9 Slot Capabilities 2, Slot Control 2, and Slot Status 2 Registers

Test TD_3_9 is now included as part of TD_1_50, as it must be run for all device/port types. The description of this test is identical to that provided for TD_1_50.



APPENDIX C. ACKNOWLEDGEMENTS

❑ AMD

- Gord Caruk
- Dean Gonzales
- Betty Luk
- Steve Manning
- Anthony Tam
- Wayne Yun

❑ Agilent

- Rick Eads
- Gordon Getty
- Thorsten Goetzelmann
- Rob Vezina

❑ IBM

- Will Atherton
- Dustin Patterson

❑ IDT

- Mike Chessin

❑ Intel

- Dan Froelich
- Manisha Nilange
- Akshay Pethe
- Marc Wells

❑ LeCroy

- Linden Hsu
- David Li
- Joseph Schachner

❑ LSI

- Richard Solomon

❑ NVIDIA

- Raymond Barrios
- Steve Glaser
- Bill Simms
- Mark Taylor

❑ PLX

- Nagamanivel Balasubramaniyan
- Swapnajit Mitra
- Satish Venugopal

❑ QuestTech

- Jerry Kinsley
- Rich Minter
- Dan Neal

❑ Tektronix

- Sarah Boen
- Jit Lim
- Steve Reinhold
- Klev Sepp